PCM

The Personal Computer Magazine for Tandy® Computer Users

Vol. III No. 7
January 1986
U.S. $3
Canada $3.95

Interfacing With Your 1000's Joystick Ports

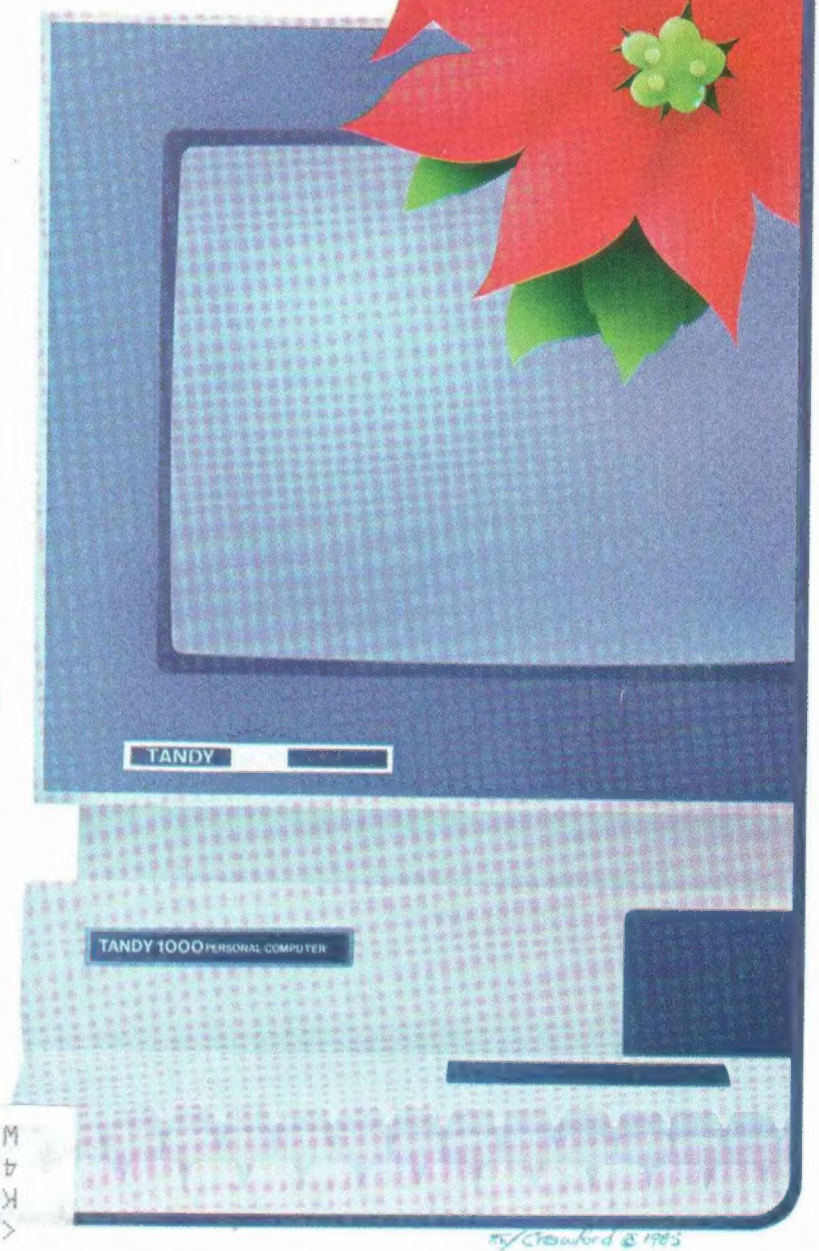
Power Databasing in Sub City

Create Superfast Video Displays

Better Security For Your Portable Files

Plus

'Turbo Powered' Graphics
Product Reviews
Bar Code Listings
and More



M. Crawford © 1985

THREE IT TAKES TWO TO TANGO

Give your Model 100/Tandy 200 total access to your desktop MS-DOS computer. Transfer files in the blink of an eye. Use the same word processing and communications commands on both computers.

Telecommuter runs on the IBM-PC/XT/AT/jr and compatible MS-DOS computers, such as the Tandy 1000/1200/2000, Compaq and AT&T 6300. TC's Fast File Transfer uses your portable's standard Telcom program, so you don't have to add anything more.

Enter text on your portable, transfer the file to your desktop and finish the job — with big computer word processing and communications power — FAST!

TELECOMMUTER™

Software that always works
Support that's always there

Only $200, MasterCard and VISA accepted.

SO SIMPLE

SIGEA SYSTEMS, INC.

19 Pelham Road, Weston, MA 02193 (617) 647-1099

Basic.

Unfortunately, a lot of Word Processing Software is not what it seems. It is not easy. But it is expensive.

Leading Edge wants to change the way you think about Word Processing.

By making it easy *yet* inexpensive.

Quite simply, we've taken advantage of all the features of personal computers like the IBM® Tandy® and Leading Edge® PCs and made them appealingly simple to use.

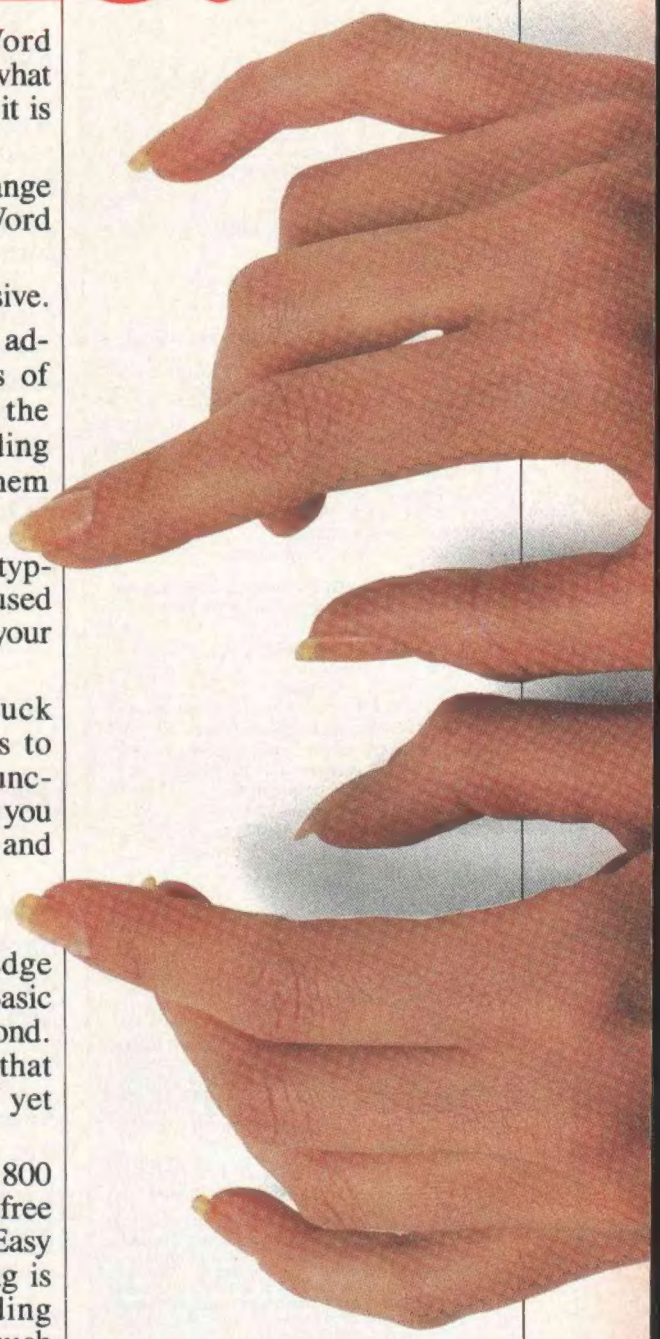
Even if you're a two-finger typist. Even if you've never used word processing before in your life.

Easy. Simple. A snap. Duck Soup. There are no codes to memorize. No complex functions. And the system guides you all along the way, in plain and simple English.

What's the catch?

There is none. Leading Edge Software Products go from Basic to Brilliant. And then, Beyond. With the kind of support that everybody else promises, yet rarely delivers.

To find out more, just call 800 523-HELP and ask for our free brochure. It's easy to get. Easy to read. The only hard thing is understanding how Leading Edge Products can do so much for you, so fast, for so little.



LEADING EDGE®
Word Processing

Leading Edge Software Products, Inc.
21 Highland Circle, Needham Heights, MA 02194
1-800-523-HELP, (617) 449-4655

PCM

The Personal Computer Magazine for Tandy® Computer Users

January 1986 Vol. III No. 7

Editor and Publisher
Lawrence C. Falk

Managing Editor Danny Humphress
Submissions Editor Jutta Kapfhammer
Copy Editor Tamara Renee Dunn
Reviews Editor E. Monica Dorth
Editorial Assistants Wendy Falk, Judi Hutchinson, Shirley Morgan
Technical Assistants John R. Curl, Ed Eilers
Contributing Editors William Barden, Jr., Randy Graham, Jim Hawk, Richard White
Consulting Editors Belinda C. Kirby, T. Kevin Nickols

Art Director Tracey Jones
Designers Heidi Maxedon, Kevin Quiggins, Sandra Underwood
Production Assistant Cindy Jett

Chief of Typography Debbie Hartley
Typography Services Jody Doyle, Angela Kapfhammer, Suzanne Benish Kurowsky

Falsoft, Inc.

President Lawrence C. Falk

General Manager Patricia H. Hirsch

Editorial Director James E. Reed
Asst. Editorial Director Jutta Kapfhammer
Executive Editor Courtney Noe
Creative Director Jerry McKiernan
Manager of Public Relations Wayne Fowler

Administrative Assistant to the Publisher Judy Brashear
Asst. General Manager for Finance Donna Shuck
Bookkeeper Diane Moore
Advertising Accounts Beverly Taylor
Dealer Accounts Judy Quashnock
Accounts Payable/ADP Lisa Ragan
Director of Fulfillment Services Bonnie Frowenfeld
Asst. Dir. of Fulfillment Services Sandy Apple
Asst. Customer Service Manager Beverly Bearden
Customer Service Representative Monica Wheat
Word Processor Manager Patricia Eaton
Chief of Production Services Melba Smith
Dispatch Janice Eastburn
Research Assistants Laurie Falk, Robin L. Long, Sharon Smith

Advertising Coordinator Doris Taylor
Advertising Assistant Debbie Baxter
(502) 228-4492
For PCM Advertising and Marketing Office Information, see Page 98

PCM — The Personal Computer Magazine for Tandy® Computer Users is published every month of the year by FALSOFT, INC., The Falsoft Building, P.O. Box 385, Prospect, KY, 40059. Phone (502) 228-4492. **PCM — The Personal Computer Magazine for Tandy® Computer Users** and the **PCM** logotypes are registered ® trademarks of FALSOFT, Inc.

Second class postage paid Prospect, KY, and additional offices. USPS N 713-470 (ISSN 0747-0460). POSTMASTER: Send address changes to PCM, P.O. Box 385, Prospect, KY 40059. Forwarding Postage Guaranteed.

Subscriptions to **PCM — The Personal Computer Magazine for Tandy® Computer Users** are $28 per year in the United States. Canadian rates are U.S. $35. Surface mail to other countries is U.S. $64, air mail U.S. $85. All subscriptions begin with the next available issue.

Payment accepted by VISA, MasterCard, American Express, Cash, Check or Money Order in United States currency only.

Limited back issues are available. Please see notice for issues that are in print and their costs. Payment accepted by VISA, MasterCard, American Express, Cash, Check or Money Order in U.S. currency only.

Contents

Features

Departments

Reviews

Cover illustration copyright© 1985 by Fred Crawford

Tandy computers offer everything . . . almost

Check out the software that completes their systems — *filePro*®

Tandy MS-DOS Software Comparison Chart

	pfs® File/Report	dBase™ II	*filePro* 16 Smallware
GENERAL CHARACTERISTICS:			
Menu driven	yes	no	YES ✓
Allows user to create integrated business systems	no	programmer required	YES ✓
Developed systems and data can be moved to multi-user environments	no	no	YES ✓
Professional support available from the software's authors	no	no	YES ✓
PRICE	$265	$595	$495
CAPACITIES:			
Fields per record	100	32	999 ✓
Characters per record	1679	1000	4608 ✓
Records per file	1300	65535	16,000,000 ✓
Indexes per file	1	7	12 ✓
Number of digits per numeric field	20	10	24 ✓
Number of files usable concurrently	1	2	10 ✓
Files span multiple drives	no	no	up to 8 ✓
FEATURES:			
Full-screen facility for creating custom screen layouts	yes	no	YES ✓
Full-screen facility for creating custom report layouts	no	no	YES ✓
Built-in field types (error checking)	no	3	12 ✓
User-defined field types	no	programmer required	200 ✓
Conditional math	no	programmer required	YES ✓
User-defined menus	no	programmer required	YES ✓
Change file layout without losing existing data	possible	possible	automatic ✓
Data protection	no	programmer required	YES ✓
Password security	no	programmer required	YES ✓

The new Tandy computers can really speed things up. If you don't let obsolete software bog you down.

Try *filePro* 16, the latest generation of data base software from the authors of Profile®.

Choose it for its power (check our capacities); its ease-of-use (check our features); and its unique ability to grow as your needs grow, and even move the systems you create to multi-user computers.

filePro is available for the Tandy 1000, 1200 and 2000 computers (hard disk required). Available directly from us or through your Radio Shack store's Express Order Software service.

To order, call (800) 847-4740.

For customer service or when dialing in New York State, call **(212) 398-9290.** Ask for Smallware sales. Visa and MasterCard accepted.

Also available for other computers, including IBM PC, IBM PC-XT, and NCR Decision Mate.

The Small Computer Company, Inc.

✓240 **230 West 41st Street, Suite 1200, New York, New York 10036**

New or Classic — We've Got the Fizz!

This is an especially important month for us — and this is an especially important issue, too.

As you will see, we have answered the requests of many of you and begun a service where you can get the programs listed in PCM on disk. PCM ON DISK should solve one of the biggest problems in using the fine programs in our magazine — the time it takes to type them in and debug your typing mistakes. PCM ON DISK will cover all the programs in PCM over about 20 lines — for both the MS-DOS and Portable Computers. Note, however that PCM ON DISK is readable only by the Tandy 1000, 1200, 2000 and 3000. Portable programs may be downloaded to your portable from one of these machines.

One thing you should note: PCM ON DISK consists of only the programs. You will need your copy of PCM for documentation and instructions on how to load and run the programs themselves.

But the beginning of this new service is not the only exciting thing that is happening this month. Beginning with this month's edition of PCM, you will be able to buy what so many of you tell us is your favorite resource in Radio Shack Computer Centers. It is part of a three month trial period that will run through March's issue.

For the first time that we (or anyone else) knows of, an "outside" magazine will be sold in Tandy's own stores. We're not alone, of course, two other magazines, which also support Tandy MS-DOS machines, will be in these stores, too.

One of them is new, a product of Camden Communications called *PC Companion*, and which was begun primarily to be part of this test. The other is an old timer, *80 Micro*, which has given some support in the past to MS-DOS along with its primary interest — the Models I, III and IV. If you're picking up this issue of PCM in a Computer Center, I hope you will look at the other two as well.

If you are reading this in a Radio Shack store, you will note that all advertising which appears in the issue is only in support of Tandy itself or from firms which have either hardware or software that is a part of Tandy's Express Order System. So, virtually everything you see advertised in the "Tandy" edition can be ordered directly through the Computer Center itself — and be in your hands quickly.

I feel personally comfortable with Tandy's decision to restrict the advertising which appears in its own stores. At the same time, this presented us with a problem: What about all the PCM advertisers who are not a part of Express Order.

What we did, simply, was create two versions of PCM. We call them New PCM and Classic PCM. Sound familiar? (I know it wasn't too original, but it was the best I could come up with at the time.) The *idea* was original, though — a special new PCM edition for the Tandy stores and the tried and true PCM edition for our regular distribution.

New PCM follows Tandy's guidelines and does not contain advertising that does not come either from Tandy itself or from Express Order suppliers. Classic PCM contains all the advertisments — Tandy, Express Order or non-Express Order.

We've picked up a number of new ads, as you will see if you follow PCM. Many of those are Express Order participants and, as such, appear in both issues. To "fill" some pages in New PCM, we have added a few programs and articles from past issues — after all we have a wealth of these. We're over two years old!

This last decision was one of the most interesting ones. We could have made the magazine smaller — but felt that would have been "cheating" the people who bought through the Radio Shack stores. Or, as others have, we could have given away some free ads to fill pages, but felt that would have been "cheating" our paid advertisers and allowing some degree of misrepresentation. So, there are some "reprints" in this month's New PCM to help new readers catch something of the flavor of our magazine.

As I mentioned, this is set up as a three month test by Tandy. If you like being able to purchase PCM in your local store, please let Tandy know. It may become a permanent fixture. However, you'll notice that we also have a subscription card enclosed with this issue. We hope you'll subscribe. If you do, we'll send you a copy of Classic PCM to your home or office each month. And, I think you'll find, like many thousands of others already have, that PCM is the most useful peripheral your Tandy 1000, 1200, 2000, 3000 MS-DOS or Portable 100, 200 or 600 Computer can possibly have!

Before closing this month's column, I want to wish the Tandy 1000 a Happy Birthday. The 1000 turned One on November 8 and there was much to-do in Fort Worth to celebrate.

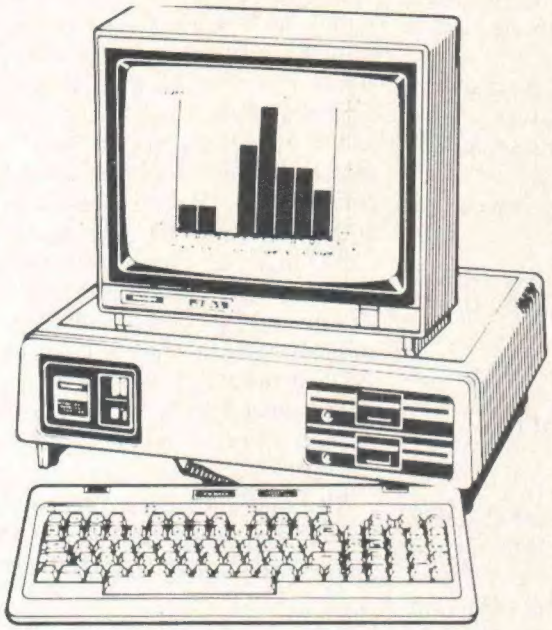
We should all celebrate, too. The 1000 is, dollar-for-dollar, the best single buy in the MS-DOS field. May it continue to grow for many more years.

— Lonnie Falk

NOCONA

ELECTRONICS

TRS-80 Model 2000 by Tandy®



TRS-80® Model 100 by Radio Shack

MS-DOS Based

Computer System

256K MOD2000 W/MONO MONITOR	1319.00
256K MOD2000 W/COLOR MONITOR	1629.00
256K MOD 1200 HD 2 DR.	1499.00
128K MOD 1000 W/MONO 1 DR.	779.00
128K MOD 1000 W/COLOR 2 DR.	1119.00
256K MOD 1000 W/HARD DISK	1439.00
24K MOD 100 PORT	399.00
24K MOD 200 PORT	679.00
64K MOD 4D 2 DR	839.00
512K MOD 6000 2DR.	3099.00
512K MOD 6000 w/15MEG HD	3779.00
15 MEG HARD DISK	1119.00
35 MEG HARD DISK	2059.00
DMP 105 PRINTER	145.00
DMP 2200 PRINTER	1149.00
DMP 130 PRINTER	249.00
DMP 430 PRINTER	585.00
DMP 2100P PRINTER	1019.00
CGP 220 COLOR PRINTER	449.00
DWP 510 PRINTER	999.00
TRACTOR DWP 210	99.00
TRACTOR DWP II/510	159.00
TRACTOR FEED DMP2100P	99.00
PARALLEL PRINTER SWITCH	96.00
PARALLEL PRINTER CONTROLLER	179.00
PARALLEL PRINTER CABLE	29.00
DISK VIDEO INTERFACE	399.00
32K MOD 600 PORTABLE	1149.00
512K MOD 3000	1819.00

100% RS COMPONENTS NO FOREIGN DRIVES OR MEMORY FULL WARRANTY
ALL RS SOFTWARE 20% OFF CATALOG PRICE
CASHIERS CHECK OR MONEY ORDER MUST ACCOMPANY ALL ORDERS

(817) 825-4027

NOCONA ELECTRONICS • BOX 593 • NOCONA, TX 76255

NO, THANK YOU!

Editor:

Thank you for including our new product, *PowerText Formatter* in the October 1985 issue of PCM [Page 92]. However, you misrepresented our name. *PowerText Formatter* is available from Beaman Porter, Incorporated, 417 Halstead Avenue, Harrison, NY 10528, (914) 835-3010 or 835-3156.

Again, thank you for your consideration.

Wanda Volpe
Beaman Porter, Inc.

NOTHING WAS STIRRING

Editor:

I recently purchased a Mouse Systems mouse to use with my system. Upon installation of the mouse, I get an error statement, "Cannot initialize COM1 port." I checked with Mouse Systems and was told that their mouse will not work with the Tandy 2000.

I found a software company using this mouse with the Tandy 2000. They told me they made a software change but were not willing to tell me what the change was.

The mouse manual specifies the following configuration for the COM1 port:

8259 Input	IRQ4
I/O Address	3F8
Interrupt Number	0C
Interrupt Address	30

Note: All numbers are Hex.

I got no answers from the local Radio Shack people. Can anyone direct me to someone who could help me configure COM1 as above or initialize it?

Phillip LaRose
Hallam, PA

WHERE THERE'S A WILL

Editor:

A recent magazine review of *DeskMate* for the Tandy 1000 stated that wide spreadsheets could not be printed. This is correct, but there is a simple solution. The worksheet permits the copying of the spreadsheet onto the disk as an ASCII file. This file can then be read into the Text program of *DeskMate* and printed. There is no difficulty printing 132 characters wide by any length.

The exact procedure is:

F7	Select, to select the entire spreadsheet;
F8	Copy, to move the worksheet into the buffer;
F8	Copy, to initiate the disk write; you can use the same filename for the Text files as was used for the Worksheet file — they will have different extensions so they won't conflict;
F12	To return to the main menu, where you will see your worksheet is listed under Text;
ALT-F6	To reset your printer specs to 132 columns.

W. Aird Flavelle
Calgary, Alberta

GIVE 'EM CREDIT

Editor:

The advertisement for your MS-DOS special interest group on Delphi (Group MSDOS) in the November PCM is misleading. You must have a credit card (MasterCard, Visa or American Express) to get a permanent user name online. This was not stated in the advertisement.

E.J. Allison
Fullerton, CA

It is true that this point caused a bit of confusion among those taking advantage of our free Delphi sign-up offer. The Delphi ad in this issue mentions the deposit/credit card requirement.

In order to start your permanent user ID, Delphi requires a cash deposit of $20 or a credit card number.

We have also included Delphi's toll-free customer service telephone number in the revision of the advertisement. If you are having trouble logging on or using the system, a helpful Delphi representative can be reached at (800) 544-4005.

Canadian subscribers, please note there is a $3/hour surcharge for Datapac.

END OF FILE

Editor:

I bought my Tandy 1000 on the first of July and I just now received my first copy of your magazine. Although I only have Tandy's 1.01 version of BASIC, I'm going to start typing the *Sketch* program described in your October issue. Hopefully, when I finish, I'll find out where to get Version 1.03.

I tried using Tandy's *DeskMate* for a full page editor for assembly language code, and it worked very well as long as I didn't type anything in the first column. But after the sixth time I used it, I got ? END OF FILE ENCOUNTERED ON INPUT FILE error statements. The error statements don't seem to cause trouble, but do you think I should be doing something different to eliminate them?

My many attempts at using EDLIN to make batch programs bring another question. More often than not, after the batch program ends I have two MS-DOS prompts on the screen. I end the batch programs with a CTRL-Z. What should I do to eliminate the second prompt?

I'm looking forward to many hours of fun and education on my 1000 and to your next issue of PCM magazine.

Walt Stecker
Garden Grove, CA

The version number you referred to is for the Tandy 2000 — not the 1000. The current version of BASIC for the Tandy 1000 is 2.02. Your local Radio Shack Computer Center or Plus Computer Center should be able to get an upgrade for you.

The problem you're having with DeskMate *is most likely due to an erroneous EOF (end of file) character within your file (ASCII 10). Most programs will not read past this character.*

The extra MS-DOS prompt on batch files baffled me for quite a while. Finally, I discovered that if I put the CTRL-Z as the last character on the last command line of the file, instead of on a line by itself, the extra prompt was eliminated.

Come join us . . .

PCMfest℠

Palo Alto — *Feb. 14-16*

You're invited to PCMfest, a great show that's just for your Tandy computer. Sponsored by PCM, *The Personal Computer Magazine for Tandy Computer Users*, it's a wonderful way to meet and exchange information with those who share your interest in the new generation of Tandy computers.

You'll also discover the greatest variety of products ever for your computer because all of the favorites and a lot of new ones will be featured in the exhibits of PCM advertisers. Try out that new program and take it home that very day!

Along with other PCM readers, you'll meet the top national experts on your computer, including those who write for or who are written about in PCM. They will answer your questions on the spot.

PCMfest also will include a comprehensive lineup of free seminars on topics of immediate concern — and all of them designed to help you get the most out of your Tandy computer.

The Hyatt Hotels—Palo Alto will be offering special rates ($71, single or double room) for PCMfest. The show opens Friday evening with a 7 p.m. to 10 p.m. session. It's a daytime-only show Saturday — the exhibits open at 10 a.m. and run continuously until 6 p.m. On Sunday, the exhibit hall opens at 11 a.m. and closes at 4 p.m.

Tickets may be obtained directly from PCM. We'll also send you a special reservation form so you can take advantage of the special room rate. Come to PCMfest and let's celebrate the new generation of Tandy computers!

RAINBOWfest Your admission to PCMfest also entitles you to attend RAINBOWfest, the highly popular show for the Tandy Color Computer which will run concurrently with PCMfest at the same location.

Free Seminars

Sam Redmon — *The Future of Portable Computing*
As co-founder of the Dallas-based Portable Computer Support Group, Sam markets a wide variety of software and hardware for the Tandy portables.

Bill Barden — *Assembly Languages*
Respected author of 30 books on various computer subjects, Bill will speak on one of his specialties, assembly language programming.

Steve Bjork — *User Interfaces*
Owner of SRB Software and author of numerous commercial software products, Steve will speak on creating better user interfaces.

Howard Wolowitz — *Database Management*
President of Small Computer Company, Howard is one of the developers of Small's popular *Profile* and *filePro* database management systems.

Danny Humphress — *ViaNet — Tandy's Intra-Office Communications System*
PCM managing editor and owner of a consulting firm in Louisville, Kentucky, Danny will take away the mysteries of using Tandy's new ViaNet networking system.

YES, I'm coming to Palo Alto! I want to save by buying tickets now at the special advance sale price.

Please send me:

_____ three-day tickets at $9 each — total _____

_____ one-day tickets at $7 each — total _____

(Circle one) Friday Saturday Sunday

Handling Charge $1.00

TOTAL ENCLOSED _____

(U.S. Currency Only, Please)

☐ Also send me a hotel reservation card for the Hyatt Hotels-Palo Alto ($71, single or double room).

(please print)

Name _____

Address _____

City _____ State _____

Telephone _____ ZIP _____

Company _____

☐ Payment enclosed, OR ☐ Charge to:

☐ VISA ☐ MasterCard ☐ American Express

Account Number _____

Exp. Date _____

Signature _____

Make checks payable to: PCM. Mail to: PCMfest, The Falsoft Building, P.O. Box 385, Prospect, KY 40059. To make reservations by phone, call: (502) 228-4492.

Advance ticket deadline: Feb. 7, 1986. Orders received less than two weeks prior to show opening will be held for you at the door. Tickets will also be available at the door at a slightly higher price. Children under four, free; four and over, full admission price.

Becoming a 'Power User' with Telecom

By Bobby Ballard

One of the most valuable features in *DeskMate* is the Telecom section. This section allows you to access the outside world of information and communicate with others. When you then couple this ability with Delphi and the MS-DOS group sponsored by PCM you become a real power user.

If you haven't taken advantage of your PCM subscription and signed up for your free lifetime membership to Delphi, then take this opportunity to do so and get used to using the Telecom feature of *DeskMate*. You can do it all at the same time as we go through it here this month.

Telecommunicating is easy but it can be intimidating, especially to the new user, and sometimes even the experienced user can get lost with all the features and equipment on the market. Using Delphi as our example, let's go through the features in the Telecom section and see if we can't make sense of all those cryptic codes and jargon.

The only items you need in order to use the Telecom section are *DeskMate*, a communications port and a modem. There are many ways to set up your system with these tools. You might add a communications port that allows for an external (outside the computer) modem, or you have the option of installing an internal (inside the computer) modem that doesn't require any cables other than the direct-connect phone line.

Once you've set up your system with the necessary equipment, boot up *DeskMate* and select the Telecom section by moving the window over and pressing ENTER. You will notice that unlike the other sections of *DeskMate*, Telecom does not prompt you for a filename but displays filenames of your autolog files in the window. It immediately displays the communications parameters available plus a new menu at the bottom of the screen.

(Bobby Ballard is a free-lance writer and the owner of a computer software and consulting firm. He also operates a BBS in Brooklyn. Bobby can be contacted at 1207 Eighth Avenue, Apt. 4R, Brooklyn, NY 11215.)

If you don't select an autolog file, *DeskMate* displays the Telecom status menu. If you do select a filename when entering Telecom, then you immediately send into autodial mode for the service you selected.

What's It All Mean?

Let's stop and take a look at what all those status settings mean at this point and talk about setting them for logging onto Delphi. If you're new to telecommunicating, don't worry, you'll be an expert in a few minutes. If you have some experience in communicating, you may still want to stay with me here as some of the features set up in *DeskMate* are confusing and are covered only briefly in the manual.

When you communicate using your computer as a terminal, you use ASCII code to accomplish the transfer in most cases. This is true with Delphi, other information utilities like CompuServe and just about all BBSs. ASCII code, put simply, is a set of agreed upon binary codes that represent the keys on your terminal, plus a few other signals for controlling communication, which we will cover at another time. For now, just remember the way your terminal and the host, such as Delphi, know what you're saying to each other is through the use of ASCII code. By the way, ASCII stands for American Standard Code for Information Interchange.

Looking at the Telecom status screen in *DeskMate*, notice first the new menu on the bottom of the screen. Press F11 and you will see the same alternate menu that is accessible from the main menu of *DeskMate*. Press F11 again to get the new Telecom menu back.

The first selection on the Telecom status menu, F1, will always set the communications status or parameters back to their original settings that come with *DeskMate*. So, if you get things all out of whack while experimenting, don't worry; F1 will take you back to the original settings. Just remember the settings you have when you exit Telecom to go back to the main menu of *DeskMate* will be stored and used the next time you enter Telecom unless you select an autolog file.

The next menu item is Select (F2) and works like the

ENTER key to change the settings on the status screen. Try this: Place the blinking cursor over 110, using the arrow keys to move it around, and then press F2. You see the status change from 300 Baud to 110. The change is denoted with the setting blocked out in highlight. Now move the cursor to 300 and press ENTER this time. The same thing will happen again, making 300 the setting.

The Autolog feature under F3 is used to invoke an Autolog file. You must first edit or create an Autolog file using the next key, F4, for Editlog. Both of these features have additional menus and screens and we will discuss them in more detail.

Pressing the F5 key will take you directly to Terminal mode for communicating. This is where the action is and where you spend all your time while online. Press F5 and take a look at this screen and its accompanying menu. To exit the terminal mode, press F12 and you will be back in the Telecom status menu. I will go into more detail about Terminal mode and the features found there.

The F6 menu selection in Telecom status mode will clear the buffer. By pressing F6, any information you have captured in the receive buffer of *DeskMate* will be cleared and lost forever, unless you did a save first, which is the next item on the menu.

To save information in your receive buffer onto a diskette for future use, press the F7 key and answer the prompt with the name of the document you wish to save. Be sure to use the extension of .DOC so it will show up in your Text window on the main menu of *DeskMate*. Once you have saved your buffer, you may then clear it by using the F6 key discussed above.

The F8 key allows you to print information in your buffer on a line printer if you have one hooked up and installed. Just press this key and your printer will print everything in the buffer. If you need to change the printer parameters before doing the printout, use the ALT F6 key. After you set up your parameters, press F12 and then use the F8 key to start printing.

If you are going to send a file to someone else or to PCM, then you will need to use the next feature: Load. To use this you press F9 and then answer the prompt that asks for which filename to load with the name of the file used to save the document. Be sure your file is less than 17,000 bytes in length or it won't fit in the send buffer. We will talk about the buffer in more detail when we cover Terminal mode.

The last menu selection, F10, allows you to see the information in your buffer. By pressing this key, you can scroll through the buffer and read information just captured, or check to see that the information you are about to send is all intact. One way to keep all of this straight in your mind is to remember that the last five selections (Clear, Save, Print, Load and Display) refer to the buffer. So, just remember that Clear, for example, is really Clear buffer, and so on.

On this same screen you see all of the status settings for communicating. Under Current Status there is a list of 10 settings that may be changed or reset to match the different hosts you will be calling. These settings effect the parameters and protocols used to make sure that you and the host are able to understand each other.

The first selection tells *DeskMate* whether you have an autodial or manual dial modem. If your modem can autodial, then select "YES" and you will be greeted with a Define Modem Type screen. Here is where you must know some of the features of your modem and how to program it for autodial. To get back to the Status menu press F12. I will cover setting up for autodial later as it is somewhat complicated and requires a thorough understanding of your modem.

If you have a manual modem, then leave the selection for Autodial Modem on "NO." Of course, if you have an autodial modem, you may still dial manually whether you are set for autodial or not.

The next selection is for setting the Baud rate to match the modem and host you are calling. First, if you have a 300 Baud modem, then leave the setting as it comes from Tandy. If you have 1200 Baud or 2400 Baud, and the host you will be calling can handle that speed, then make the selection you desire. For Delphi, you may use 300 or 1200 Baud.

The next setting, Data Word Length, determines the length of the actual ASCII code you will be using. When calling Delphi, I've found that either will work well, but I got better results using 7 bits on my terminal. However, Delphi suggests you use 8 bits. If you are getting some garbage on the screen when you make connection, then change it to see if that clears up the problem. Sometimes it's impossible to know in advance which setting will work best. This is true when calling around to different BBSs. Once you know, though, you can put the information in an autolog file and never have to worry about it again.

The parity setting is for determining whether a simple error checking protocol will be used. For many services, Even is the setting you will be using. The recommendation for Delphi is None. Without going into a lot of detail, let me just explain that the parity setting determines how many bits in each byte will be set of each character you type or send. If the parity is set to Even then each character you send is set with an even number of bits in each byte. This is a very simplified explanation of a simple error checking protocol.

The next setting is related to the above two settings. This setting is for determining how each character you send and receive are delimited. If you set it for 2 stop bits then the system will look for two bits to be set at the end of each character sent and received. Most of the time you will leave this setting at 1 stop bit.

The XON/XOFF Flow Control parameter is used to control the way your terminal and the host keep up with each other. If you are getting characters too fast, then *DeskMate* will send a control character to tell the host to "wait" so your terminal can catch up. Then, when your terminal is caught up and ready to receive more information, it will send another control character to tell the host to resume sending data. Setting this feature "on" will cause the above protocol to take effect. Otherwise, data is sent and received without any receive buffer control.

The next parameter on the list, ASCII Character Filter, tells *DeskMate* to print only those characters that represent letters and numbers, plus a limited set of the control characters such as line feed, carriage return, form feed and backspace to name a few. All other characters in the ASCII character set, like NULL, BELL and CANcel will be ignored. This is especially useful for capturing data to a

printer as you are receiving it. Most of the time you will leave this setting off and then switch it on when you are printing data while online. This will give you a clean printout of data as it is being received.

The Line Feed Filter, next down the list, controls whether line feeds are stripped or left in as text is being sent. If this setting is on, then all line feeds are ignored and text is displayed accordingly. If switched off, then all line feeds are sent to the screen and/or printer. You will have to experiment with different settings depending on the service you are calling and whether you are using a printer while online. For most calls you place you will want to leave this "off."

The Echo feature determines the duplex setting. Echo and Duplex are interchangeable concepts in telecommunications. When Echo is "on" then each character you type is echoed locally on your screen. This is the same as half duplex. When this setting is "off" you are in full duplex mode. This means that the host must echo back your keystrokes in order for you to see what you have typed. For Delphi and most other hosts you will leave this off. However, if you call a service and get double characters on your screen when you type, then switch this setting to eliminate this problem. If you place a call and can't see what you type then switch this setting to "on" and your keystrokes will be echoed locally for you to see.

The last setting is for those of us with autodial modems. This tells *DeskMate* how many times to retry connecting with a particular service before giving up. This is especially useful for calling BBSs that are always busy. Using the Redial capability allows you to do other things while your computer does the dialing. This is a valuable feature to have and can save you lots of time and trouble trying to make a connection.

Autodialing

To set up your copy of *DeskMate* for autodialing, move the cursor to "yes" after the Autodial Modem selection at the top of the Telecom status screen. Now press ENTER or F2 to make the selection. This will take you to a new screen for setting up the parameters your modem requires to autodial. Since I can't know all the different codes for the different brand modems, I must use the one I own as an example; the Modem II from Tandy.

To set up for dialing other computers, select F2 when you are in the Define Modem Type screen. Please note that the only way to get to this screen is to go to the Autodial Modem selection in the Telecom status screen, place the cursor over "yes" and press ENTER or F2. This seems inconvenient, but you only have to set up this section once for your modem and then it's always there.

After selecting F2 to set up for computer dialing, you must enter the codes used to actually program the modem. To create this file you use the keystroke macros listed at the bottom of the screen. Instead of having to type the commands in by hand, you just use the function keys to put the commands on the screen. For example, strike F7 to put Delay on the screen since it's the first command in the autodial file. You will see "Delay:" followed by a blinking cursor for the delay amount. For the Modem II, I get best results with a value of nine. You might try other values here though to see what works best for you. The available values are one through 10 and you must use zero to represent 10.

Following are the autodial settings I use for the Modem II. You can just type them in using the function keys,

HARD DRIVE PRICE BREAKTHROUGH

STARTING AT $299.95!

5 to 60 megabytes complete easy to install internal and external systems
— will boot just like an XT — fully warranteed —
same day shipping from stock — fast service — free technical assistance

$499.95 to $999.95
STREAMING OR FLOPPY TAPE BACK UP SYSTEMS
complete internal and external systems

$1,995.00 — IBM XT WORKALIKE SPECIAL
20 megabytes — one floppy — 640K — clock calendar — serial port — parallel port
monochrome card — monochrome monitor with tilt and swivel base
Keytronics 5151 style keyboard — one year warranty — shipped from stock

MEGADISK™ HARD DISK DRIVE SYSTEMS **TOLL FREE ORDERING 1-800-343-8841**

For the { **IBM/PC**, [illegible], **TRS/80 Models I/III/IV/4P**, Compaq, PC Workalikes, **Color Computers**, Heath/Zenith, Max/80
Complete with Hardware, Cables, Software and Quikfit Installation

5 Megabytes Internal Mount	starting at	$ 279.95
10 Megabytes Internal Mount	starting at	439.95
20 Megabytes Internal Mount	starting at	579.95
5 Megabytes External System	starting at	499.95
10 Megabytes External System	starting at	639.95
20 Megabytes External System	starting at	799.95
Tape Backup System — Interna; Or External	starting at	449.95

Wow! New Low Prices

DOS Systems Available: { IBM/Heath — DOS, 1.0, 2.0, 2.1, 3.0, or later
TRS/80-LDOS, TRSDOS 6.x, Newdos/80, Dosplus, CP/M, COCO DOS, Max/80 LDOS, OS9

FULLY WARRANTIED — PARTS AND LABOR — 24 HOUR SERVICE — CALL TOLL FREE — 1-800-343-8841

IBM ENHANCEMENT CARDS

Multifunction with 384K Serial port, parallel port Clock calendar, game port	$259.95	Color graphics	$139.95
		Monochrome/RGB Color	229.95
Serial port — two port	69.95	Disk controller with cable — up to 4 drives	79.95
Monochrome only	119.95	384K memory	129.95
Monochrome graphics w/parallel port	159.95	Hard drive controller	159.95
		Keytronics 5151 style keyboard	129.95
		Network card	249.95

DEALER INQUIRIES INVITED
CALL 1-617-872-9090

PRICES CHANGE EVERY DAY
CALL 1-800-343-8841

PC © 1985 **SOFTWARE SUPPORT, INC.**

1 Edgell Road, Framingham, MA 01701 (617) 872-9090 Telex-383425

Hours: Mon. thru Fri. 9:30 am to 5:30 pm (E.S.T.) Sat. 10 am to 3:30 pm

SERVICE POLICY — Our Professional Technical Staff Is Available To Assist You Monday Through Saturday.
WARRANTIES — Up To One Full Year Parts And Labor. Floppy Disk Drive Power Supplies — Five (5) Years.
SERVICE — 24 Hour Turn-A-Round On All In-Stock Parts. Dealer Inquiries Invited. Call 617-872-9090

Please Call For Shipping, Handling And Insurance.
Cash Discount Prices

Toll Free 1-800-343-8841
Please Call For Our Latest Price Saving Specials.

Not Responsible for Typographical Errors. Prices and Specifications May Change Without Notice.

followed by the values I've used. Then press F12 to escape this menu and save your work.

```
DELAY: 9
SEND: *ODT
RECEIVE: T
NUMBER:
SEND: X
RECEIVE: X
WAITC
```

The above file will work with the Modem II, but you may have to put two asterisks in the second line if you are using the file after just cutting on the Modem. Try it and see what works best for you. Remember, this will *not* work at all if you don't have your communications parameters in the Telecom status screen set for 8 bits. The Modem II can only be programmed in 8 bit mode. So, it will only autodial if your status settings are at 8 bits. This true for the Modem II, but not necessarily true for other modems.

Let's Call Delphi

In a later column I will go through how to set up autolog files so you don't have to go through the following, but let's manually do so at first so you can get online and get involved. Also, this is the method that is used to determine how to build an autolog file. If you don't know what prompts are coming from the system, you must go through a manual logon and store the information to use in building your autolog file.

First, set your status with 8 bits, no parity and 1 stop bit, and set the modem Baud rate. The autodial feature should already be set. Then press F5 to enter the terminal mode. Now you have the option of dialing the number by hand or using your autodial feature to dial the number for you.

You can use one of several numbers to access Delphi. These numbers are for data networks that provide you with quality lines and avoid the cost of calling long distance. There is no charge for using these special data links.

To use Uninet, call 800-821-5340 for a local access number. The number for finding your local Tymnet number is 800-336-0149. In Canada you can use the DATAPAC network. To find out the local access number for DATAPAC call 617-491-3393.

If you are autodialing the number, then type F8 to enter the phone number from your keyboard and follow the number with ENTER. The Modem will autodial the number and give a "Connection Good" prompt at the lower-right corner of the terminal screen.

To dial without autodial, just dial the number you are going to use, place the modem in originate mode and hang up the phone. Now you are connected to the network through which you will be accessing Delphi. Next you must use the sequence of codes for your particular network to reach Delphi. The following gives the sequence for all three networks.

On Uninet, after making the connection, type ENTER period (.) ENTER. You will then get a prompt for "Service:" which is for the service you will be accessing. Type in GVC and press ENTER. That's it, you are then connected to Delphi through the Uninet network.

For Tymnet, start by typing A at the first prompt, which is "please type your terminal identifier:" in most cases. When you get the prompt "Please log in:" type in the word DELPHI, or type V3035, and press the ENTER key. You are now on Delphi.

On DATAPAC in Canada it makes a difference as to the Baud rate you are using. For 300 Baud, type one period (.) followed by the ENTER key. For 1200 Baud, type in two periods (..) and ENTER. Next you type SET 2:1, 3:126 and press ENTER. Finally, type p 1 3106, DELPHI and press ENTER. That should do it.

The rest of the sequence for logging onto Delphi is the same regardless of the network you are using. The first prompt you see is for your username, and here is where you type the name under which you are registered. If you are not registered, type PCMSUB or PCM ORDER and press ENTER. If you have a PCM subscription use the first response, if not and you wish to subscribe, use the second response. After you are registered you will type in your own username that you register under.

The next prompt is "PASSWORD:". Here you type in your password that you use if you are already registered. If you are not and you are a subscriber, type in your subscription number found on the mailing label of your latest issue. If you are signing up for a subscription, type SENDSUB at the password prompt. Now you are completely logged in and ready for a session online, or if you are registering for the first time, just answer all the questions to complete your registration. It's easy, and the connect time is not charged to you while you register.

After you have registered with Delphi, you will be contacted by phone and mail to verify your account. Usually, by the next day or two you are able to log back on using your own username and password. On your second call you are taken through a series of questions that help determine the type of terminal you have and the service you require.

I want to point out that when you place this second call and you are setting up your terminal type, do not tell Delphi that you have an 80 character line length even though you do. This goes for any system that you log onto that asks for line length or column width. You should use the number 79 so as to get single spaced text. Answering with 80 will cause the host to send 80 characters before sending a line feed or carriage return. Hence, when the end of the line is reached by *DeskMate* it will automatically generate a line feed. This will be followed by an additional linefeed from the host, which will cause your text to double space.

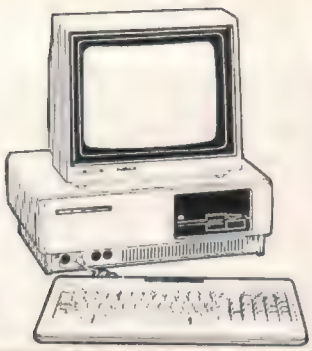
Send Us A Note

Now that you know how to call and access Delphi using *DeskMate*, use the Text feature of *DeskMate* to compose a letter or note to PCM and send it along using the Telecom section. It's easy to do and you'll have fun in the process. It's a great way to inexpensively get a grip on telecommunications, too!

After you've composed the message, just load it into your Telecom buffer by using the F9 Load command from the Telecom status menu. Then, once you are online and in the MS-DOS group, send a message and use the file in your buffer when Delphi prompts you for the text of your message.

I will cover this aspect of using *DeskMate* more in the future and also will discuss creating autolog files to put you right into the MS-DOS group without having to do all this work. Of course, both of these tasks will make your life easier and will save you time and keystrokes as well. In addition, I can't think of a better way to get answers to your questions than by contacting PCM using *DeskMate* on Delphi. You get personal attention and a quick answer to boot! PCM

PERRY COMPUTERS



CALL
1-800-248-3823

COLOR COMPUTERS		LIST PRICE	OUR PRICE
26-3127	64K Extended Color Computer 2	$219.95	$ CALL
26-3134	16K Standard Color Computer 2	119.95	CALL
26-3129	Thinline Disk Drive 0 for Color Comp	349.95	CALL
26-3018	Extended Basic Kit	39.95	36.00
26-3030	OS-9 with Editor Assembler	69.95	59.50
26-3012	Deluxe Joystick	29.95	25.00
26-1208	CCR-81 Recorder	59.95	50.00
	VIP Integrated Library	149.95	139.00
	VIP Writer	69.95	59.00
	VIP Calc	69.95	59.00
	Telewriter Disk		59.00
	Botek Interface		57.00
TANDY 2000			
26-5103	Tandy 2000 Two Disk	$1599.00	$1225.00
26-5104	Tandy 2000 One Disk & 10 Meg HD	2499.00	1885.00
26-5111	Monochrome Monitor	199.95	165.00
26-5112	Color Monitor	599.95	510.00
26-5140	Hi Res Graphic Monochrome Board	299.95	255.00
26-5141	Hi Rec Color Graphics Chips	124.95	105.00
26-5160	Internal 128K Memory Board	179.00	150.00
26-5300	Lotus 123	495.00	420.00
26-5311	Microsoft Multiplan	195.00	165.00
26-5320	Framework	695.00	590.00
26-5352	dBase II (Data Base)	595.00	420.00
26-5330	Multmate	249.00	205.00
MODEL 200 COMPUTERS			
26-3860	24K Model 200 Portable Comp.	$999.00	$ CALL
26-3866	24K RAM Mem. Exp. Chip for 200	249.95	210.00
26-3804	AC Adaptor	5.95	5.00
26-3805	Accoustic Coupler	39.95	34.00
26-3816	8K RAM Memory Expansion Chip	119.95	95.00
26-1409	Printer Cable	14.95	12.70
26-1410	Modem Cable	19.95	17.00
26-3809	Briefcase	49.95	42.50
26-3811	Soft Carrying Case for 100/200	39.95	34.00
26-1183	Bar Code Reader	99.95	85.00
26-3806	Disk Video Interface	499.00	415.00
26-3829	Multiplan ROM for 100	149.95	127.50
MODEL 4 COMPUTERS			
26-1070	Model 4D 64K 2 Drive Computer	$1299.00	$ 895.00
26-1122	64K Memory Expansion	69.95	65.00
26-1127	Model 4 Internal Drive 1	399.95	385.00
26-1123	Model 3 to Model 4 Upgrade Kit	399.00	385.00
26-1084	Model 4P Modem Board	149.00	95.00
26-1085	Model 4P Travel Case	24.95	21.25
26-1134	Model 4 Hard Disk Kit for 15meg	79.95	67.95
26-1530	Multiplan Model 4	199.00	169.00
26-1595	Super Scripsit Model 4	199.95	169.00
26-1635	Profile 4 Model 4	249.95	200.00
26-2216	CP/M Plus Model 4	149.00	127.00
26-2231	Double Duty Utility	69.95	58.00
MODEL 12 & 6000 COMPUTERS			
26-6021	Model 6000 512K 2 Drive Comp.	4499.00	3125.00
26-6022	Model 6000 512K 15Meg HD	5499.00	3795.00
26-6014	Model 12 to 6000 Upgrade	1595.00	1325.00
26-6015	Model 6000 512K Memory Board	1095.00	905.00
26-6019	Model 6000 256K Memory Kit	199.95	185.00
26-6017	Card Cage for Model 12	199.00	169.00
26-6052	DT-100 Data Terminal	795.00	600.00
26-4155	15 Meg Hard Drive Primary	1995.00	1355.00
26-4171	35 Meg Hard Drive Primary	2995.00	2095.00
26-4157	Installation Kit for Primary HD	349.00	265.00

RADIO SHACK PRINTERS		LIST PRICE	OUR PRICE
26-1276	DMP-105 Dot Matrix Printer	$ 199.00	$ 169.00
26-1275	TRP-100 Portable Thermal Printer	299.00	230.00
26-1280	DMP-130 100 cps Tri. Mode Printer	349.00	285.00
26-1268	CGP-220 Color Ink-Jet Printer	699.00	595.00
25-1257	DWP-220 Daisy Wheel	599.00	460.00
26-1277	DMP-430 24 Wire Matrix Printer	899.00	745.00
26-1270	DWP-510 43 cps Daisy Wheel	1495.00	1225.00
26-1274	DMP-2100P 24 Dot Wire Matrix	1495.00	1195.00
26-1279	DMP-2200 HiSpeed Matrix Printer	1695.00	1440.00
26-1269	PT-64 Printer Controller	249.95	210.00
26-1498	SW-302 Printer Switch	119.95	100.00
26-1477	Auto Sheet Feeder for DWP 510	499.95	420.00
TANDY 1200			
25-3000	Tandy 1200 One Disk & 10 Meg HD	$1999.95	$1525.00
25-3010	VM-3 Green Monitor	219.95	185.00
26-3212	CM-2 Color Monitor	459.95	390.00
25-3040	Monochrome Display Adapter	219.95	185.00
25-3043	Graphics Display Adapter	299.95	255.00
25-3044	Graphics Master	695.95	540.00
25-3061	Captain Multifunction Board	599.95	475.00
25-3130	MSDOS/BASIC	89.95	76.50
25-3170	Wordstar Professional	395.95	335.00
25-3160	Framework	695.95	590.00
25-3161	PFS File	140.95	119.00
25-3190	dBase III	695.95	590.00
TANDY 1000			
25-1000	Model 1000 128K 1 Disk Drive	$ 999.95	$ 705.00
25-1001	Model 1000 256K 10Meg HD	1999.00	1495.00
25-1005	Disk Drive Expansion 1000	199.95	170.00
25-1006	RS-232C Interface 1000/1200	99.95	85.00
25-1007	Hard Disk Control Board	299.95	255.00
25-1011	Memory Plus Board	319.95	255.00
25-1013	Modem 1200 Option 1000/1200	299.95	230.00
25-1025	10 Meg Hard Disk	699.95	559.00
25-1501	MS-DOS Reference T-1000	34.95	29.00
25-1502	BASIC Reference T-1000	34.95	29.00
26-3211	VM2 Green Monitor	149.95	125.00
26-3212	CM2 Color Monitor	459.95	390.00
EPSON/COMREX PRINTERS			
20001015	Homewriter 10	$ 288.00	$ 230.00
20001025	LX-80 Dot Matrix Printer	299.00	250.00
20001035	FX-85 Dot Matrix Printer	499.00	385.00
20001040	JX-80 Color Dot Matrix Printer	699.00	485.00
20001050	HI-80 4 Pin Plotter	599.00	390.00
20002010	RX-100 Dot Matrix Printer	499.00	400.00
20002030	FX-185 Dot Matrix Printer	699.00	525.00
20003010	LQ-1500 18 Pin Head with Intf	1495.00	1095.00
20003510	CR-IIE Comrex Com Riter	499.00	295.00
20001510	FX-80 Tractor Feed	39.95	34.00
20004080	NLQ Parallel Intr	209.95	175.00
20001515	LX-80 Tractor Feed	49.00	40.00

All prices and offers may be changed or withdrawn without notice. Advertised prices are cash prices. C.O.D. accepted ($10.00 charge per carton on C.O.D. Call for further C.O.D. information.) M.C., Visa, AX, add 2%. All non-defective items returned will be subject to 10% restocking fee. Defective items require return merchandise authorization. Call for R.M.A. Number before returning. Delivery is subject to product availability.

For Technical Questions and Information on our complete line of computer accessories and current prices.

CALL 1-517-625-4161
FOR ORDERS ONLY CALL 1-800-248-3823

Mon., Wed. & Fri. 9-9, Tues. & Thurs. 9-6, Sat. 9-3

124 S. MAIN ST, PERRY, MICH. 48872

A little machine language and a lot of BASIC team up to create a feature-packed database sysem

'DB11' The Super Store, Search and Sort System for your Computer

Part I

[NOTE: Because of the length of the module listings creating the DB11 database system, they will be included with part two next month.]

By Robert D. Covington

When I originally entered into the world of computers, someone told me that a computer can help organize data and eliminate paper work. Obviously, that person never owned a computer. As I look around my office, I have two walls covered with file cabinets and book cases full of all the paper my computer has generated.

When I started collecting all this data, I had a very hard time organizing and cataloging all the information. One day, when trying to find an article in *80-Micro* (oh, those dismal pre-PCM years), I decided to let my computer help me organize all my information.

At first, it seemed like a good idea. Then, as I looked around for a database program that could handle all my data, I was astonished to find that the databases usually cost more than my computer! Since at the time all I needed was a simple non-relational database that could keep track of columns of information, I decided to write a small database on my Model I in BASIC.

(Bob Covington has been a computer programmer and consultant for the past six years, most recently focusing his attention on both the Model 100 and the 2000. He is also a technical writer and editor. Bob can be contacted at P.O. Box 37007, St. Louis, MO 63141.)

The resulting program was a nice little program that handled any database requiring less than 256 bytes of data per record. Unfortunately, I had an application that required 257 bytes (or something like that). Luckily, at the same time, I purchased my Tandy 2000.

The Tandy 2000 not only allowed me to break the 256 byte barrier in BASIC, but also allowed me to add a fairly lengthy set of extensions to the original database. The resulting program is listed in the next few issues of PCM and is titled *DB11* (I know it is not a very original name).

DB11 Definitions and Concepts

Probably the most important term you should understand at this point is the word "database." Put simply, a database is an organized collection of data. A database is much like a file folder containing many sheets of paper about one general topic. Usually, databases are used to store information that can be organized into common elements of information. For example, a mailing list might contain information that can be broken down into common elements of name, address, city, etc.

In most database systems, these common elements of information are called "fields." A field can be compared to a column in a table of information

While fields can be thought of as the columns in a table of information, records are more like rows. A record includes one set of information from each field in the database. In a mailing list database, a record would unite a single name with its perspective address, city, state and ZIP code.

One extra division used by *DB11*, which is not common with many other database systems, is the concept of a "sub-database." A sub-database is a "logical" database that contains a portion of the fields in the main (key) database. For example, in a mailing list database containing name, address, city, state and ZIP code fields, a sub-database could be created that just included the name, state and ZIP code fields. In this sub-database, the fields for address and city would be temporarily invisible to the user. This is not to say the data is not there, but just not accessible under the current sub-database. That is why a sub-database is called a "logical" database; it logically extracts and manipulates information from the physical main (key) database. The concept of the sub-database should become clearer as we progress.

DB11 Features and Capabilities

A database system such as *DB11* is used to collect information into these three logical data divisions and allow that information to be manipulated in a variety of ways.


TANDY 600 PORTABLE COMPUTER
ESC
TAB
CTRL
SHIFT
ENTER
SHIFT
ALT
NUM

The new disk-based Tandy 600.

A breakthrough in portable computing power and versatility.

Finally—a portable that matches the performance of a desktop computer. Our new Tandy 600 features a 16-bit microprocessor, an 80-character by 16-line display, a built-in 3½" disk drive that stores 360K of data and 32K RAM (expandable to 224K*).

Five Resident Applications

With the Tandy 600's larger display and expanded memory, Multiplan spreadsheet analysis can hold more information. Word processing is as easy as using MS-Word. You'll have quicker access to documents, and better storage with the built-in disk drive. File is an electronic data base for names and addresses, expenses, client billing, inventory and more. And you can keep a large number of different files on the pocket-size diskettes. With Telecom and the Tandy 600's built-in modem, you're able to communicate with other computers over phone lines and access national information networks. Telecom will even dial the phone number of anyone listed in the File program. Calendar helps you keep track of daily tasks and activities.

The easy-to-learn resident System Manager lets you run each application, exchange information between applications and manage the files created. And you can add BASIC/ROM (26-3904, $129.95) to write your own programs.

Tandy . . . Clearly Superior™

If you've been looking at portable computers, compare them with the Tandy 600 (26-3901, $1599). You'll be amazed at what it can do!

Available at over 1200 Radio Shack Computer Centers and at participating Radio Shack stores and dealers.

Radio Shack®
COMPUTER CENTERS
A DIVISION OF TANDY CORPORATION

Send Me a Free Tandy 600 Brochure.

Radio Shack, Dept. 86-A-135
300 One Tandy Center
Fort Worth, TX 76102

Name _____
Company _____
_____
Address _____
City _____
State _____
Zip _____
Phone _____

* RAM upgrades are in banks of 96K each (26-3910, $399.95). Prices apply at Radio Shack Computer Centers and participating stores and dealers. Multiplan and MS-Word/TM Microsoft Corp.

DB11 is designed to be an easy to use, powerful and fast non-relational database system to run on Tandy's MS-DOS systems with GW-BASIC or a compatible compiler. Some of the features of *DB11* are:

- Powerful addition and editing commands.
- Provisions for adding fields to the main database after it has been created.
- Multi-field sorts.
- Multi-field searches with a variety of relations.
- Edit, new record and manual tag flags for each record.
- Multiple delete priorities.
- Extensive reporting capabilites.
- Calculations on reports.
- Single and multi-line reports.
- Sub-databases.
- Multiple reports-per-disk support.
- On-line help available.
- Capable of manipulating large files. 32,767 records; 32,767 bytes-per-record (currently set to 1024); 1,000 fields-per-record (currently set to 255); support for 255 sub-databases and reports for each database.

The only real limiting factor of the entire database system is a sub-database can only include as many fields as can fit on one screen. All this means is if you have 255 or so fields, you can not view and edit all the fields at the same time. The data must be broken down into multiple sub-databases to view sections of the whole database.

Loading and Running *DB11*

DB11 is a database system written for the Tandy 1000, 1200, 2000 and 3000 with at least one disk drive and 256K of memory. The program is designed to be run under standard GW-BASIC (the BASIC language that comes with the computer) or under the GW-BASIC compiler (more on this later).

Program 1 contains the main program, which is always resident in memory. This program handles all the file I/O and provides a set of common subroutines that can be used by the other modules. Program 1 should be entered in GW-BASIC and saved normally under whatever you want to call the database. I will refer to this module as DB11.BAS.

Before saving the final copy of DB11.BAS, Line 35 needs to be changed depending on your individual system. This line contains nine variables that define the foreground and background colors for your particular monitor. Line 35, as listed, contains the default colors for a monochrome Tandy 1200, 2000 or 3000. These values will work for any computer, but you will probably want to change them for your own system. Table 1 contains an explanation for each color variable and my suggested values for a Tandy 1000 and Tandy 2000. If you do not like my suggestions for colors, a listing of colors and attributes can be found in your system's BASIC reference manual under the COLOR instruction.

Program 2 contains the database format functions overlay, which is used to create, change, delete and list information on databases and sub-databases. Program 2 should be entered under GW-BASIC and saved with the command SAVE*d*:DBFORM.BAS",A where '*d*' is the drive that contains Program 1.

Program 3 contains the main database overlay. This module is responsible for loading all the other overlays and for adding, editing and viewing information in the database. Program 3 should be entered under GW-BASIC and saved with the command SAVE*d*:MAIN.BAS",A where '*d*' is the drive that contains all the other overlays.

Once all three programs have been entered and saved, two batch files need to be created in MS-DOS. From the MS-DOS prompt, type:

COPY CON *d*:CDBASE.PTR ('*d*' is drive letter)
00 ENTER
00 ENTER
^Z (CTRL-Z) ENTER

This creates a default CDBASE.PTR file that tells *DB11* the system is starting up for the first time. The second file is a batch file that will load and execute *DB11*. To create this file, type the following at MS-DOS:

COPY CON *d*:DB11.BAT ('*d*' is drive letter)
BASIC DB11 /F:4
^Z CTRL-Z

All of the files that are used in the database discussed so far should be on the same disk.

To run *DB11*, change the current default drive to the drive that contains the database (via the CD command) and type DB11. The system will then display a few things on the screen and finally stop at the "Database Format Functions" menu (if this is the first time the system is started).

Creating a Database

Once the system is up and running, the first thing you must do is define a database to work with. *DB11* is very flexible with the types of information that can be stored in a database. For demonstration purposes, however, let us create a simple mailing list database.

To create a new database, select the 'M' (Make Database) option on the "Database Format Functions" menu. After entering this option, the program asks for the database name. The database name is an eight-character word that identifies the database and all of its associated files. Since the database name is used as part of the filename of the database files, make sure the name is unique (unless you want to overwrite an old database) and that it abides by MS-DOS's rules for legal filenames. In our example case, we will call the database MAIL (do not forget to press ENTER after every prompt).

The program will then ask you which drive to place the database on. The drive letter can be any legal drive that contains a formatted disk (you can even use the disk the program is stored on). In our case, we will assume there is a formatted disk in Drive A and use it to store the database on (enter A at the prompt).

Next, the program will ask how many records should be allowed in the database. This question was added to allow

Table 1
Color/Attribute Variables Set in Line 35

Variable	2000	1000	Description
BU	0	7	Background for input and view prompts.
BI	7	9	Background for deleted records and edit.
BD	0	0	Normal text background color.
BE	7	9	Background color for error messages.
FU	9	0	Foreground for input and view prompts.
FI	9	7	Foreground for deleted records and edit.
FD	7	7	Normal text foreground color.
FH	10	10	Foreground high intensity.
FE	0	7	Foreground color for error messages.

you to limit the size of a database. This is especially useful when there are multiple databases on a single drive. In our case, we do not need to limit the number of records so enter 32767 — the maximum *DBII* will allow. If you happened to choose a lower number and decided to increase it at a later time, one of next month's modules allows the maximum record limit to be changed.

The above questions define the existence of the database.

DBII now prompts for information about the type of data you want to store in the database.

The first question in this series asks for the field name. The field name is a unique 15-character word that identifies a particular field. In our demonstration mailing list, type NAME for the first field.

The program then asks for the valid field type for data that is to be entered into this field (eight are listed). The most commonly-used field type is alphanumeric. This data type allows any non-control character to be entered into a field as data. The alpha only type (2) allows only letters and periods to be entered into the field. The automatic time (3) and date (4) stamps tell the computer to load the current time or date into the field when a record is saved. This is very useful in determining when a record was last updated. If these types are chosen, the computer does not allow the user to enter any characters into the field.

Data Types five through seven are used to describe number-only fields. The only difference between the data types is the amount of space used to store the number and the precision (number of significant digits) that are to be recognized and stored.

The last data type, date entry (8), is used to identify fields that contain dates. For demonstration purposes, define the name field as an alphanumeric field (1).

If a field type was chosen that uses letters, *DBII* asks if lowercase letters should be allowed to be entered. If you answer this question with an 'N', the computer will automatically convert any lowercase letters to uppercase before storing them in a field. On the other hand, if you reply with a 'Y', both upper- and lowercase letters may be stored in the field. For the NAME field, let's allow only uppercase letters, so reply with an 'N'.

The last question about the field asks the number of characters that should be allowed to be entered for the field data. The input length can be any length under 80-characters. For the NAME field, we will use 20 characters (enter 20 at the prompt).

Next, the computer asks if the information was entered correctly. This prompt is a little different from the others in that it does not require that you press ENTER after the character (like the menu prompts). Unless you made an error, hit the 'Y' key only.

Now the computer will ask if there are any more fields in this database. Since there are, press 'Y'.

The above procedure should now be repeated for the rest of the fields in our demonstration database. Below is a list of the fields that needs to be entered and the information that describes them.

Field Name	Type	Lowercase OK	Length
Address	1	N	20
City	2	N	15
State	2	N	2
ZIP Code	6	—	5
Date stamp	4	—	—

Once the database has been created, we now have to create at least one sub-database to provide a logical view of the data. To do this, press B from the database format functions menu to invoke the "Build Sub-Database" option. Once you have entered this option, the computer asks for the name of the sub-database. This name *is not* a file-name and can contain any information that is needed to identify the sub-database you are going to create. For demonstration purposes, enter Main at this prompt.

Once you have done this, the computer clears the screen and invokes the full screen editor. While in the editor, you can edit data on the screen in much the same way you would with a word processor.

Table 2 contains a complete listing of all the control characters and their associated functions. The main purpose for the full screen editor is to create a "screen" where the data in each record of the database can be entered and displayed.

For demonstration purposes, we will create a screen that includes all of the fields in the database. First, press the HOME key to position the cursor in the upper-left corner of the screen. Now type Name of person followed by a colon (:) and 20 periods. The colon followed by the periods tells the computer where the name field is defined on the screen. It is very important that the number of periods used match the length of the field data.

Next, type two spaces and the word Address. This time, instead of entering the colon and 20 periods for the field data, we will have the computer do it for us. To do this, press the PG DN key until the address field appears on the lower line of the screen.

To enter the data for this field, press the TAB key (or CTRL-I) and the field data will appear on the screen. Continue this process for all the fields until your screen looks like Figure 1. When you are done, press CTRL-Q and answer the exit confirmation question with a Y.

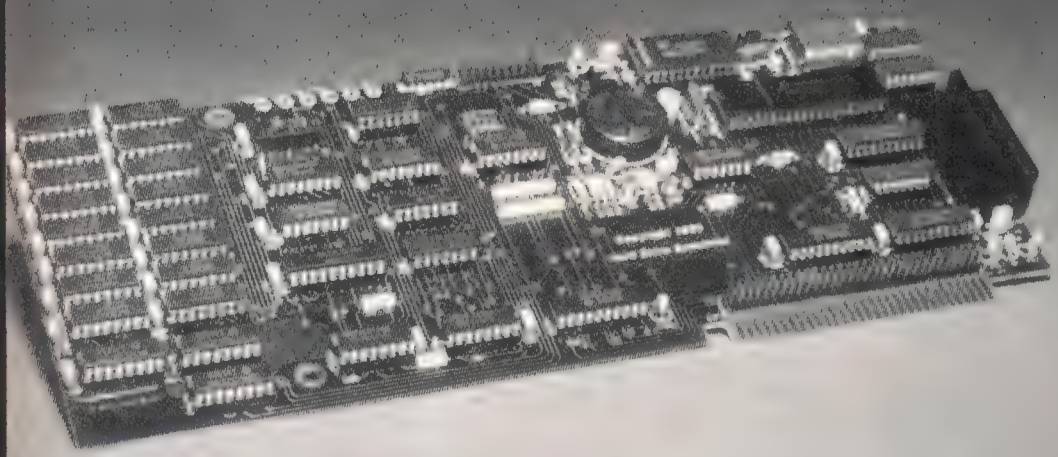
Next, the computer asks you to select the field that belongs in the section preceded with an inverse colon. In the case of our first field, the field that should be selected to go into the first "blank" is the Name field. To select this field, use the up and down arrows until Name appears on the screen. Then, press the space bar to select the field. This process should be repeated for all the fields defined in the first sub-database.

After all the fields are defined, the

Table 2
Full Screen Editor Commands

CTRL-D	Duplicate last character for rest of line.
CTRL-E	Erase current line and index lower lines up.
BKSP	Backspace and delete.
HOME	Move cursor to upper left corner.
END	Move cursor to lower right corner.
CTRL-L	Insert line at current row.
ENTER	Move to begining of next line.
CTRL-Q	Quit editing screen and return.
INSERT	Insert space at current cursor position.
DELETE	Delete character at current cursor position.
ESC	Erase (blank) current line.
Arrows	Move cursor in direction of arrow.
SHIFT Arrows	Move to extreme position in arrow direction.
PG UP and PG DN	Scan through field information (field help).
CTRL-I	Insert periods for current help field.
TAB	Same as CTRL-I

The Ultimate Expansion for the Model 1000!
TanPak ™ by Hard Drive Specialist



7 Functions Include:
DMA
Serial Port
Memory up to 512K
Clock/Calendar
Printer Spooler
Memory Disk
Expansion Port for Future Options

Features Include:
Gold Edge Cards
Pre-Tested and Burned In
Full Documentation
Supporting Software
One Year Warranty
Expandable to 512K

TanPak ™ PCM CERTIFICATION SEAL

The TanPak ™ expansion board has been designed to allow expansion beyond the scope of the standard Model 1000. Seven of the most needed functions/features have been combined into one package using only one expansion slot. Your remaining spaces are left free for future expansion needs.

Seven Function/Features on One Board

512K Memory Expansion
Socketed and expandable to 512K. This is done by two banks of memory using either two 64K increments (128K), or 256K increments (256K or 512K). This allows a total of 640K in the Tandy 1000.

Serial Port
Using the same configuration as the Model 1000 port you are assured of complete compatibility as well as being able to configure it as COM1 or COM2.

Clock-Calendar
Quartz-controlled for a high degree of accuracy, featuring a battery backup.

DMA
The DMA (Direct Memory Access) is used on the Model 1000's first memory card. It increases memory speed and is a must for hard drive operation.

Printer Spooler
Use part of your TanPak ™ memory as a printer buffer. Choose the amount of buffer space you need and stop waiting on your printer.

Memory Disk
Use part of your TanPak ™ memory as a RAM drive. With a solid state drive you can store, retrieve, and sort data quickly and easily.

Expansion System
The TanPak ™ was designed with an expansion port that will allow upgrading with additional features when they become available. Some of the possible features are: a second parallel port, a second serial port, mouse, hard drive port, bubble memory, A/D, and D/A as well as many others.

TanPak ™ 0K . . . . . $329.

Call for pricing on memory modules.

TanPak ™ Secondary
If you already have a Model 1000 memory board and do not wish to replace it, the TanPak ™ Secondary is for you. It features Memory up to 256K, Clock/Calendar, Serial Port, Printer Spooler, and Printer Buffer.

TanPak ™ Secondary 0K . . . . . $249.

TanTel ™
Internal 300/1200 baud Modem for the Tandy 1000. Supported by the DeskMate software package.

TanTel ™ . . . . . $249.

Ordering Information

Use our W.A.T.S. line to place your order via Visa, MasterCard, or Wire Transfer. Or mail your payment directly to us. Any non-certified funds will be held until proper clearance is made. COD orders are accepted as well as purchase orders from government agencies. UPS ground is our standard means of shipping unless otherwise specified. Shipping costs are available upon request. Tandy and Radio Shack are trademarks of the Tandy Corporation. TanPak, and TanTel are trademarks of Hard Drive Specialist, a division of Compukit Corporation.

Hard Drive Specialist
16208 Hickory Knoll
Houston, Texas 77059
1-713-480-6000
Order Line
1-800-231-6671

Megabytes for Mini-Bucks

Tandy 1000 Hard Drives

by Hard Drive Specialist

Shop and compare. Hard Drive Specialist has been building hard drive systems for years and have sold thousands of subsystems to satisfied Radio Shack/Tandy customers. We use the latest state-of-the-art disk drives and controllers. Our drives all use buffered seek logic and plated media to result in almost one-fourth the average access times found on our competitors drives. Plated media results in longer platter life and high resistance to head crashes not found on coated media drives. We utilize high quality construction throughout including heavy duty power supplies and gold edge card connectors. Internal drive systems include an interface card and a half-height hard drive that replaces the top disk drive in both size and power consumption. External units include an interface card, case, power supply and hard drive unit. All units require a memory board with DMA such as the TanPak™ multifunction card, or a Tandy Board, Part #25-1004.

Hard Disk System Features:

- All sizes rated after formatting.
- One secondary drive may be added.
- Error checking and correcting controller.
- Software drivers included.
- Buffered seek drives for improved access time.
- Built in power up diagnostics.
- Plated media for long disk life.
- Gold connectors used throughout.
- 1 year warranty.
- Boots directly from Hard Drive.
- Uses Tandy 1000 MS DOS.

10 Megabyte Internal	**$ 549.**
20 Megabyte Internal	**$ 749.**
42 Megabyte Internal	**$1349.**
10 Megabyte External	**$ 699.**
20 Megabyte External	**$ 899.**
42 Megabyte External	**$1495.**

Ordering Information

Use our WATS line to place your order via Visa, MasterCard, or Wire Transfer. Or, mail your payment directly to us. Any non-certified funds will be held until proper clearance is made. COD orders are accepted as well as purchase orders from government agencies. UPS ground is our standard means of shipping unless otherwise specified. Shipping costs are available upon request. Tandy and Radio Shack are trademarks of the Tandy Corporation. TanPak, is a trademark of Hard Drive Specialist, a division of Compukit Corporation.

Hard Drive Specialist
16208 Hickory Knoll
Houston, Texas 77059
1-713-480-6000

Order Line
1-800-231-6671

Figure 1

Name of person: ..... Address: .....
City: ..... State: ..... Zip Code: .....
Date stamp: .....

program should return back to the data format functions menu. Once it does, go back into the build sub-database option and make a sub-database with just the Name, ZIP Code, and Date Stamp fields like the one in Figure 2. Name this sub-database `Logical View 1`.

After the Mail database and its two sub-databases (Main and Logical View 1) have been created, exit the database format functions menu and return to the main menu (option R).

Adding Information

Now that the database has been defined, the next step is to select a sub-database for entering information. To do this, press `C` from the main menu to select the change sub-database option. Next, use the up and down arrows to scan through the list of available sub-databases until the desired sub-database name is on the screen. Since we are going to be entering information, select the sub-database Main since it contains all of the available fields in the database.

Now that you have defined and selected a database and sub-database, you are ready to enter data. To do this, choose the A option from the main menu to invoke the add data to database option.

After a few moments, the screen will contain the data you entered for the Main sub-database screen except that all the periods have been replaced by highlighted prompts (underlines or inverse video depending on your color values). These highlighted prompts are the areas where you can enter data for each field.

Figure 2

Name: .....
ZIP Code: .....
Date Stamp: .....

After the entire screen has been printed, you are automatically placed into the input entry editor. This editor allows you to enter information into each of the fields in the current sub-database.

To make entry easier, the input editor contains a set of command keys that are described in Table 3. The most important of these command keys are probably the up and down arrows. In the input editor, these arrow keys allow you to move to the next or previous field. If you move past the last record, the current record is saved and a new record is started.

Another way to save the current record and start a new one is to use the CTRL-E. This key is very useful when a database contains many fields and all the entries are made on the first few fields. With this key, you do not have to press ENTER or the down arrow keys repeatedly to move past the last record to save the current record.

CTRL-R and CTRL-D will duplicate field data. CTRL-R is used to duplicate the information in the last field in the current field. For example, if for some strange reason, a persons name was the same as his address in our sample mailing list, CTRL-R could be used to duplicate the name field data into the address field. CTRL-D, on the other hand, is used for duplicating the previous contents of the current field into the field. For example, if when adding information to the sample mailing list database you find that all the ZIP codes are the same, you can use CTRL-D to recall the previous contents of the ZIP code field from the last record.

Table 3 contains a few more standard command keys, which I will not describe here. Once you have entered a few records of information and tried out some of the input editors commands, hit CTRL-Q to quit the input editor and exit back to the main menu.

To view or edit the information you just entered, choose the 'E' option to invoke the edit/view mode. Like in the entry mode, the edit mode prints the current sub-database screen and fills in the periods with highlighted prompts. In this case, however, the highlighted prompts contain the data that you entered into the database. While in edit mode, there is a large set of command keys available. These are summarized in Table 4.

When you enter the edit/view mode, the first available record is displayed on the screen. On the bottom of the screen, you will see a line telling which record you are viewing and the conditions of various flags and modes.

Starting on the left, the first flag is the tag flag. This is a manually set flag that will allow you to tag certain records for special identification. Tagging is useful when you want to make a report that contains just a few random records. In order to toggle the tag flag for the current record, press the space bar.

In addition to individual tags, the `M` key will tag all records that meet the current find parameters (set by the `F` key). This function could be used in our sample database to tag all records that contained ZIP codes under 1000. Any record that does not meet the find parameters is left alone. This means if a record is tagged before the `M` key is pressed, it will be tagged afterwards. Sometimes, however, you do not want to keep all the old tags currently in the database. To clear these tags, press the `C` key.

The next flag from the left is the edit flag. This flag is automatically set by the system when a record is edited. The edit flag is very useful when you want to make a printout of all the records that have been modified since the last printout. This allows you to check if all the edit changes were made correctly. The edit flag for the current record can be toggled with the `Y` key and all edit flags can be cleared with the `W` key.

The last flag the system stores is the new record flag. This flag is automatically set when a record has just been entered into the system. Since all of your entries are new, this flag should be set for all the records in the system. Like the edit flag, this flag is useful for

Table 3
Advanced Input Routine Key Definitions

CTRL-D	Duplicate field data from previous record.
BKSP	Backspace and delete.
Left Arrow	Backspace and delete.
DELETE	Backspace and delete.
CTRL-D	End entry on current record and save.
HOME	Move to first field on screen.
ENTER	Move to next field.
Down Arrow	Move to next field.
CTRL-D	Abort current record and exit edit/entry mode.
CTRL-R	Replicate data in previous field in current field.
ESC	Erase (blank) current field data.
Up Arrow	Move to previous field.

Table 4
Edit Mode Commands

Commmand	Line No.	Explanation
Q or CTRL-Q	4300	Exit edit/view mode.
H	4400	Help.
J	4500	Set jump value.
V	4600	Set access level.
D	4700	Set delete level.
Up Arrow	5000	View previous record that qualifies.
Down Arrow	5100	View next record that qualifies.
Left Arrow	5200	View previous record.
Right Arrow	5300	View next record.
HOME or SHIFT Up	5400	Move to beginning of file.
END or SHIFT Down	5500	Move to end of file.
PG UP	5600	Jump back set number of records.
PG DN	5700	Jump forward set number of records.
G	5800	Go to specific record.
E	6000	Edit current record.
A	6100	Add record to end of database.
DELETE	6200	Delete record.
INSERT	6300	Unkill record.
U	6400	Update time stamp.
Y	6500	Toggle edit flag.
Z	6600	Toggle new record flag.
W	6700	Clear all edit flags.
X	6800	Clear all new record flags.
F	7000	Set find paramaters.
M	7300	Mass tag of records matching find.
Space	7400	Toggle tag of record.
C	7500	Clear all tags.
S	7600	Toggle search view mode.
T	7700	Toggle tag view mode.

checking new changes to the database. The new record flag for the current record can be toggled with the Z key and all new records flags can be cleared with the X key.

The last two statuses report if *DBII* is in normal view mode or in tag and/or find modes. Normally, moving between records is performed by the up and down arrows. In this mode, the only records that are skipped are deleted records. When tag mode is invoked, however, only records with tags are displayed. In other words, all untagged records are skipped. Pressing T will toggle tag mode on and off. Likewise, when find mode is invoked, only records meeting the current find parameters (set by F) are displayed. Pressing S will toggle tag mode on and off. Both of these view modes are very useful in editing large files.

Unfortunatly, while this database is fairly fast, using the up and down arrow keys to view large databases is very time consuming. To help speed up finding particular records, a few commands have been added to provide faster access to records.

To view the end or beginning record of the file, the END and home keys respectively are used. In addition, if you like using arrow KEYS, the SHIFT up and SHIFT down arrows also move you respectively to the beginning or end of the file (the SHIFT arrow keys only work on the Tandy 1000 and 2000).

The PG UP and PG DN keys respectively jump back or forward a set number of records. Initially, these keys will jump in increments of 10 records. To change this value, press the J key and change the value to whatever jump off set you desire.

The left and right arrow keys respectively display the previous and next record regardless of the mode. These keys are especially useful in find and tag modes where the up and down arrows might skip a few records. These keys are also very useful in displaying deleted records. If a record has been deleted, the record is displayed in inverse video (or the equivalent on your monitor).

The last important view command, G (for goto), allows any record to be accessed by entering its record number. This function is very useful when edit changes need to be made in random places in the database.

Speaking of edit changes, I have yet to describe any of the edit commands in this section. The most important edit command, as you might expect, is the E key for editing the current record. When you press the E key, *DBII* invokes the input editor and allows you to edit the data in the current record using the commands in Table 3.

When making edit changes, you will notice that any edit changes you make are in a different color or attribute from normal. This is done to emphasize any edit changes made, allowing you to review them before saving the record to disk. If you accidentally enter edit mode or you destroy the data in the record beyond repair, the edit can be aborted with CTRL-Q and nothing will be updated on disk. If you wish to make your edit changes permanent, either press CTRL-E or move beyond the last field.

If, while viewing data in the database, you need to add a record or two, this

The publishers of PCM are taking an interest in a different type of programming

VCR

THE HOME VIDEO MONTHLY

That's right. We've decided that programs like *Super Utility* and *Sketch* aren't the *only* great ones around. There are also *Silver Streak* and *Star Trek* and *Some Like It Hot* — software of a different sort.

That's why we are introducing **VCR, The Home Video Monthly**, the magazine for the new generation of home viewer.

Home video has evolved beyond the "hacker" era, when you needed a degree in electronics just for a little entertainment. Most people don't care about how the signal-to-noise ratio and wow-and-flutter specs of their equipment compare to the latest models. They simply want to know how best to use and enjoy the equipment that they have.

And that is what **VCR** will offer — how to get the very best in home entertainment from your equipment.

Each month, **VCR** will bring you previews and ratings of every new offering on tape and disc: music videos, children's shows, how-to guides, and movies, movies, movies.

We will tell you which shows the critics themselves will be taping on the networks and cable, along with tips from the experts on how to get the best possible reproductions. And you can turn to us for the answers to your questions, ranging from the trivial to the technical.

Even more, each month we will feature exclusive interviews with the stars and the star-makers, along with articles designed to help you relive some of your fondest video "memories" of the past.

Yes, programming is more than spreadsheets and databases. And we know you'll want to be a charter subscriber to *the* guide to the very best in entertainment software.

VCR, The Home Video Monthly, debuts in January 1986.

Yes, I'm ready for some real entertainment! Send the next 12 monthly issues of VCR to my door.

Subscribe now for only $15 and save 36% off the regular newsstand price.

Name _____

Address _____

City _____ State _____ ZIP _____

☐ My check in the amount of _____ is enclosed. (In order to hold down costs, we do not bill.)

Charge to: ☐ VISA ☐ MasterCard ☐ American Express

Account Number _____ Exp. Date _____

Signature _____

*Subscriptions to **VCR** are $15 a year in the United States. Canadian rate is U.S. $22. Air mail rate elswhere is U.S. $60. All subscriptions begin with the current issue. Please allow 5 to 6 weeks for first copy. Kentucky residents add 5% sales tax. **U.S. currency only, please.** In order to hold down non-editorial costs, we do not bill.

Mail to: VCR, The Falsoft Building, P.O. Box 385, Prospect, KY 40059

can be performed using the A key. Like the 'A' option from the main menu, this command key will invoke the input entry editor, clear all fields and allow you to enter information into the database. When the data is stored on disk, it is automatically stored at the end of the database. In addition, to simplify viewing and adding information in the editor, the view record position is not affected by an add.

The last edit commands I will discuss here are the delete command keys. Deleting data under *DB11* is quite different than with most database systems.

The simplest way to delete information is to move to the record you wish to delete and press the DELETE key. Likewise, if you wish to undelete a record, use the PG UP and PG DN keys to move to the deleted record you wish to recover and press the INSERT key. For 90 percent of the people using *DB11*, that is all the information needed to work with deletes.

For the other 10 percent a concept called delete levels has been added. Let's say, for example, in our sample mailing list we had three reasons for deleting people. An example of three good reasons might be the person does not exist because of bad data, the person dies or you dislike the person (and find some pleasure in deleting him or her). In each of these three cases, you have different priorities for deletes.

In the first case, you probably will never want to see an entry error so these types of deletes should have the highest priority.

In the second case, you will probably never want to see the information on someone who has died, but you might at some later time want to recall the information for a family tree or something.

In the last case, you probably would not want to permanently delete the person just in case you became friends (or at least stopped disliking each other) at a later time.

To help cope with these different types of delete conditions, *DB11* allows up to 31 different levels of delete. This means you could give typos a Level 31, deaths a Level 16 and others a Level 1.

To change the current delete level, press the D key and enter a value from one to 31. When edit mode is invoked, the default delete level is set to 16. After a delete level is set, that delete level is stored with a deleted record when the delete key is pressed.

By itself, delete levels are not very useful. To make them useful, *DB11* has a command key, V, which changes the current access level to a value between zero and 31. For example, if you wished to temporarily "ignore" that deletes with a Level 1 were deleted and include them in searches, reports and tags, you could change your access level to one. Then, when a record that was deleted with a delete level of one is encountered, the system lets it be treated as if it were not deleted. To help identify that a record has been deleted, however, *DB11* changes the color/attributes of the record on the screen to denote that it has been deleted.

Two additional commands that are implemented in the edit/view mode are the H and Q keys. The H key is used for obtaining online help about the edit/view mode. The last option, as you might expect, will exit the edit/view mode and return to the main menu. Since CTRL-Q is used so much throughout the system to exit, it too is accepted as a legal exit character in edit/view mode.

Once you have tried out all of the features of edit/view mode with the sample database, go back to the main menu, change the current sub-database to logical View 1, and return back to edit/view mode. I think if you do this, the concept of the sub-database might become a bit clearer.

General Notes About the System

As you use *DB11*, you will find that its slowest sections are loading overlays and updating screens. The delay with loading overlays is caused by the slow speed of the CHAIN MERGE command. Unfortunately, there is no standard alternative in BASIC to load overlays and keep the variable table intact. When the program is compiled, this overlay delay is almost eliminated (more on compiling next month).

The screen update delays are just about as fast as they can be in BASIC. Unfortunately, there is no standard way in BASIC or in the BASIC compiler to speed this up. As I hinted earlier, all the options in view mode are intended to help reduce screen updates.

One of the most useful features of *DB11* is its extensive error recovery logic. If the system encounters an error, and the error is recoverable, *DB11* will ask if you wish to retry, abort or ignore the error. The retry option is intended to recover errors from unavailable devices, open disk drive doors, write protects, etc. The abort option is used to offer a way of aborting back to the main menu if the error ends up being unrecoverable. If an error is aborted, the system attempts to recover as much data as it can. Lastly, if the ignore option is used, the system proceeds just as if the error did not occur. This is used mostly to save as much data as possible on a flawed disk.

Alterations

One of the original design features of *DB11* is that it is written 99 percent in BASIC. For me, this is the most BASIC code I have ever used in one program (I usually average 40-50 percent machine language in most of my programs). What this means for you is *DB11* is very easy to modify to your own needs.

Probably the most common alteration will be the color values in Line 35 of the DB11.BAS module. As I stated

earlier, I suggest you change these colors to your own liking.

Another value that will probably need to be changed is the delay subroutine starting at Line 53100 of the DB11.BAS module. This subroutine is used to pause the system while printing some error messages on the screen.

To help in any alterations, Table 5 contains a listing of the main variables used in the system. Other more technical information on the system will probably be made available on PCM's SIG on Delphi (MS-DOS).

Conclusion

That about covers all the information you will need next month to use the first three modules of *DB11*. We will explain modules 4 and 5 in the next installment. Unfortunately, since this is supposed to be an article and not a book, I have left a lot of the details about *DB11* for your own discovery, even with products like *dBase II*, which have seemingly hundreds of books describing how to use them, you still learn more by trial and error. □

Table 5
Variable Definitions

A$( )	Current record data (DB array).
AL	Current access level.
B$( )	Previous record data (DB array).
BD	Background color for normal text (usually black).
BE	Background color for error messages (usually inverse video).
BI	Background color for deleted records or edit record (usually inverse video underline).
BIOS$	Machine language MS-DOS function access routine.
BU	Background color for normal input and view modes (usually underline).
CD$	Current database name (with/drive identifier).
DF	Data found flag for search (-1 =found).
DL	Current delete level.
DR	Delete level of record.
ED	Field edited flag (128 = true).
F$( )	Field name (DB array).
FD	Foreground color for normal text (usually white).
FE	Foreground color for error messages (usually inverse video).
FH	Foreground color for "high intensity" prompts.
FI	Foreground color for deleted records or edit record (usually inverse video underline).
FM	Find mode flag (0 = off, -1 = on).

FS$( )	Status text (0 = Reset, 1 = Set).
FT$( )	Status text (0 = Off, 1 = On).
FU	Foreground color for normal input and view modes (usually underline).
HF	Status line field help field number.
JV	Jump value.
L( )	Length of field (DB array).
LD	Number of subdatases.
LR	Number of reports.
MF	Maximum number of fields in main database.
MR	Maximum record number stored on disk (top record).
MS	Maximum allowable records in database (<32,768).
ND	No keydatabase flag (-1 = true).
NR	New record flag (32 = true).
NS	No subdatabase flag (-1 = true).
PR	Previous record number.
PX( )	X coordinate of field prompt (SDB array).
PY( )	Y coordinate of field prompt (SDB array).
QN( )	Field number of search parameter.
QS	Number of parameters on search.
RD$(1,)	Database data off disk (DB array).
RE( )	Relationship between search data and DB data.
RI$	Byte stored in .INF file for each record.
RL	Record length of data pool (.DAT) file.
RN	Current record number.
RN$( )	List of report names (LR).
S$( )	Screen immage for subdatabase (24).
SC	Scan code of character most recently entered from keyboard.
SF( )	Sub-database to key database conversion array (SDB array).
SL( )	Sub-database field length during record compiling (SDB array).
SM	Maximum number of fields in sub-database.
SN	Current subdatabase number.
SN$( )	List of subdatabase names (LD).
ST$( )	Text to search for (compressed for numbers).
T( )	Field type (DB array) (1-alphanumeric, 2-alpha only, 3-time stamp, 4-date stamp, 5-integer #, 6-single precision #, 7-double precision, 8-date entry).
TG	Field tagged flag (64 = true).
TM	Tag mode flag (0 = off, -1 = on).
TP$( )	Field type descriptions (8).
UC	Uppercase lock for input.
UC( )	Upper- lowercase switch (DB array).

PCM

Rickety Downs

By Barb & Joab Jackson

Rickety Downs is a horse race Simulation game and, though it might not be the Saratoga of software, what it lacks in bells and whistles, it makes up for in exciting finishes.

The program presents a nine-race card, with Win, Place, Show and Exacta wagering on each race. The fourth and ninth races also contain a Triple.

For those of you unfamiliar with Exacta and Triple wagering, an Exacta requires you to select the first and second finishers in order; a Triple, the first three finishers in order. All payoffs are for $2 wagers, and the user is responsible for keeping his own account.

All the action takes place on the text screen and the player is prompted throughout.

The odds for each race are displayed in the lower left of the screen and range from 1:2 (lowest) to 6:1 (highest). The odds for each horse reflect that horse's chances of winning the race. You can expect the favorite to win about 35% of the time, but this does not preclude the possibility of a long shot getting his nose in front for a big payoff in any given race.

The program may be used strictly for entertainment or perhaps even a sophisticated "piggy bank." By initially seeding *Rickety Downs* with $5 and a coffee can, designating dimes as dollars and playing the races on a regular basis, your computer could well keep itself supplied with printer paper and disks. □

(Barbara Jackson is a cost analyst at Random House, Inc., and Joab is a horseman on the Maryland racing circuit. They both enjoy hiking, bird watching and computers.)

Program Comments for Rickety Downs

010-014	Title.
016-018	Initialization.
020-024	Fill race array with random numbers 1-6.
026-034	Set up screen for race.
036	Count last 20 elements of race array to determine odds.
038-050	Display odds, remove entries with odds count over six (race course is 50 units long).
052-056	Check for triple race. Flag wagering on short fields.
058-060	Wait for start.
062	Remove starting gate.
064-114	Run race using subroutines 68-108. Display winner's time.
116	If triple race compute triple payoff.
118-120	Compute exacta payoff.
122-134	Display official results and payoffs.
136-140	If races completed (9), GOTO 142. If not, wait for next race. GOTO 16 and do it again.
142	Clean up and end program.
144-148	Data for Win, Place and Show payoffs.

The listing:

```
 1Ø WIDTH 8Ø:KEY OFF:CLS:PRINT"Welcome to RICKETY DOWNS"
 12 LOCATE 4,1,Ø:PRINT"(C) 1985 by Barb and Joab Jackson":POKE 249,1
 14 LOCATE 7:PRINT"FOR AMUSEMENT ONLY":LOCATE 1Ø:PRINT"ONE MOMENT PLEASE..."
 16 CLEAR:DIM P(336),C(6),O(6),M$(6):RANDOMIZE TIMER:NR=PEEK(249)
 18 FOR X=Ø TO 6:READ M$(X):NEXT:W$="WIN":P$="PLACE":S$="SHOW"
 2Ø FOR X=1 TO 336
 22 Z=INT(RND*6)+1:C(Z)=C(Z)+1:IF C(Z)>56 THEN 22 ELSE P(X)=Z
 24 NEXT X:FOR X=32Ø TO 336:T=T+18:SWAP P(X),P(X-T):NEXT X
 26 CLS:PRINT TAB(13)"Race";NR;SPC(12)"RICKETY DOWNS";SPC(9)"Track Fast":PRINT
 28 PRINT TAB(13)STRING$(51,127)
 3Ø FOR X=1 TO 6:PRINT TAB(12) X;CHR$(29);">";SPC(48);CHR$(124):NEXT X
 32 PRINT TAB(13)STRING$(51,127):LOCATE 13,13:PRINT"ODDS"
```

```
34 FOR X=1 TO 6:PRINT TAB(12);X"  1  :1":NEXT X
36 FOR X=317 TO 336:O(P(X))=O(P(X))+1:NEXT X
38 LOCATE 14:FOR X=1 TO 6
4Ø IF O(X)=Ø THEN PRINT TAB(12);X"  1  :2"
42 IF O(X)=Ø THEN COLOR 23,Ø:LOCATE 13+X,19:PRINT":":COLOR 7,Ø:GOTO 48
44 IF O(X)>6 THEN PRINT TAB(12);X"  SCRATCHED":SC=SC+1:GOTO 48
46 PRINT TAB(12) X" ";O(X);":1"
48 NEXT X:LOCATE 4:FOR X=1 TO 6
5Ø IF O(X)<7 THEN PRINT TAB(12) X;CHR$(29);">" ELSE PRINT TAB(13)" "
52 NEXT X:IF NR=9 THEN COLOR 23,Ø:LOCATE 11,13:PRINT"Last Race":COLOR 7,Ø
54 IF SC<2 AND (NR=4 OR NR=9) THEN LOCATE 21,13:PRINT"TRIPLE WAGERING"
56 IF SC>1 THEN LOCATE 21,13:PRINT"WIN AND EXACTA WAGERING ONLY"
58 LOCATE 12,3Ø:PRINT"PRESS -S- TO START"
6Ø Q$=INKEY$:IF Q$="S" OR Q$="s" THEN LOCATE 12,3Ø:PRINT SPC(19) ELSE 6Ø
62 FOR X=1 TO 6:LOCATE 3+X,14:PRINT" ":NEXT X
64 ST=TIMER:FOR X=1 TO 316:ON P(X) GOSUB 68,76,84,92,1ØØ,1Ø8:NEXT X
66 LOCATE 11,54:PRINT USING"Time  ##.#";FT-ST:GOTO 116
68 IF O(1)>6 OR A=5Ø THEN RETURN ELSE A=A+1:LOCATE 4,12+A:PRINT" 1"
7Ø IF A=5Ø AND R=Ø THEN R=1:W=1:FT=TIMER:LOCATE 4,66:PRINT W$:RETURN
72 IF A=5Ø AND R=1 THEN LOCATE 4,66:PRINT P$:R=2:P=1:RETURN
74 IF A=5Ø AND R=2 THEN LOCATE 4,66:PRINT S$:R=3:S=1:RETURN ELSE RETURN
76 IF O(2)>6 OR B=5Ø THEN RETURN ELSE B=B+1:LOCATE 5,12+B:PRINT" 2"
78 IF B=5Ø AND R=Ø THEN R=1:W=2:FT=TIMER:LOCATE 5,66:PRINT W$:RETURN
8Ø IF B=5Ø AND R=1 THEN LOCATE 5,66:PRINT P$:R=2:P=2:RETURN
82 IF B=5Ø AND R=2 THEN LOCATE 5,66:PRINT S$:R=3:S=2:RETURN ELSE RETURN
84 IF O(3)>6 OR C=5Ø THEN RETURN ELSE C=C+1:LOCATE 6,12+C:PRINT" 3"
86 IF C=5Ø AND R=Ø THEN R=1:W=3:FT=TIMER:LOCATE 6,66:PRINT W$:RETURN
88 IF C=5Ø AND R=1 THEN LOCATE 6,66:PRINT P$:R=2:P=3:RETURN
9Ø IF C=5Ø AND R=2 THEN LOCATE 6,66:PRINT S$:R=3:S=3:RETURN ELSE RETURN
```

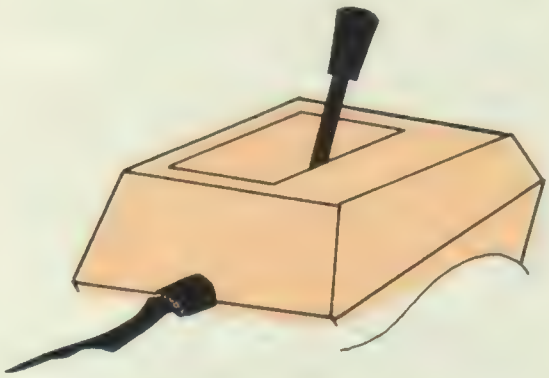
```
92 IF O(4)>6 OR D=5Ø THEN RETURN ELSE D=D+1:LOCATE 7,12+D:PRINT" 4"
94 IF D=5Ø AND R=Ø THEN R=1:W=4:FT=TIMER:LOCATE 7,66:PRINT W$:RETURN
96 IF D=5Ø AND R=1 THEN LOCATE 7,66:PRINT P$:R=2:P=4:RETURN
98 IF D=5Ø AND R=2 THEN LOCATE 7,66:PRINT S$:R=3:S=4:RETURN ELSE RETURN
1ØØ IF O(5)>6 OR E=5Ø THEN RETURN ELSE E=E+1:LOCATE 8,12+E:PRINT" 5"
1Ø2 IF E=5Ø AND R=Ø THEN R=1:W=5:FT=TIMER:LOCATE 8,66:PRINT W$:RETURN
1Ø4 IF E=5Ø AND R=1 THEN LOCATE 8,66:PRINT P$:R=2:P=5:RETURN
1Ø6 IF E=5Ø AND R=2 THEN LOCATE 8,66:PRINT S$:R=3:S=5:RETURN ELSE RETURN
1Ø8 IF O(6)>6 OR F=5Ø THEN RETURN ELSE F=F+1:LOCATE 9,12+F:PRINT" 6"
11Ø IF F=5Ø AND R=Ø THEN R=1:W=6:FT=TIMER:LOCATE 9,66:PRINT W$:RETURN
112 IF F=5Ø AND R=1 THEN LOCATE 9,66:PRINT P$:R=2:P=6:RETURN
114 IF F=5Ø AND R=2 THEN LOCATE 9,66:PRINT S$:R=3:S=6:RETURN ELSE RETURN
116 IF NR=4 OR NR=9 THEN TR=((O(W)*4)+(O(P)*3)+(O(S)*2)+(P(15Ø)*2))*2
118 IF O(W)>4 THEN EX=((5+O(P))*5)+(P(15Ø)*2):GOTO 122
12Ø IF O(W)<3 THEN EX=O(W)+O(P)+5 ELSE EX=(O(W)+O(P)+P(15Ø))*2
122 LOCATE 14,42:PRINT W:LOCATE 15,42:PRINT P:LOCATE 16,42:PRINT S
124 LOCATE 13,57:PRINT"OFFICAL":FOR DELAY=1 TO 6ØØ:NEXT DELAY
126 IF SC>1 THEN LOCATE 14,47:PRINT LEFT$(M$(O(W)),5):GOTO 132
128 LOCATE 14,47:PRINT M$(O(W)):LOCATE 15,54:PRINT RIGHT$(M$(O(P)),1Ø)
13Ø LOCATE 16,6Ø:PRINT RIGHT$(M$(O(S)),4)
132 LOCATE 18,49:PRINT USING"EXACTA    ##.##";EX
134 IF (NR=4 OR NR=9) AND SC<2 THEN LOCATE 19,49:PRINT USING"TRIPLE   ###.##";TR
136 IF NR=9 THEN 142 ELSE LOCATE 22,41:PRINT"PRESS -N- FOR NEXT RACE"
138 Q$=INKEY$:IF Q$="N" OR Q$="n" THEN 14Ø ELSE 138
14Ø LOCATE 22,4Ø:PRINT"       ONE MOMENT PLEASE":POKE 249,NR+1:GOTO 16
142 LOCATE 11,13:PRINT"**RACES COMPLETED**":POKE 249,Ø:LOCATE 22,1:KEY ON:END
144 DATA" 3.ØØ  2.2Ø  2.1Ø"," 4.ØØ  2.6Ø  2.2Ø"," 6.ØØ  3.ØØ  2.4Ø"
146 DATA" 8.ØØ  4.ØØ  3.ØØ",1Ø.ØØ  5.ØØ  4.ØØ,12.ØØ  8.ØØ  5.ØØ
148 DATA 14.ØØ  9.ØØ  6.ØØ
```

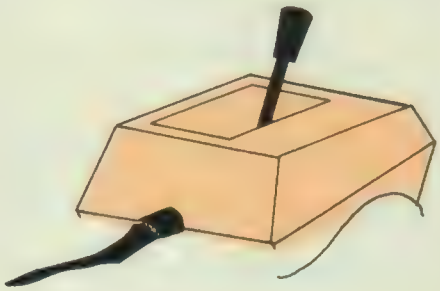
PCM

Easy techniques to get your 1000 in touch with the "real world"

Interfacing with the Tandy 1000's Joystick Ports

By William Barden, Jr.
PCM Contributing Editor





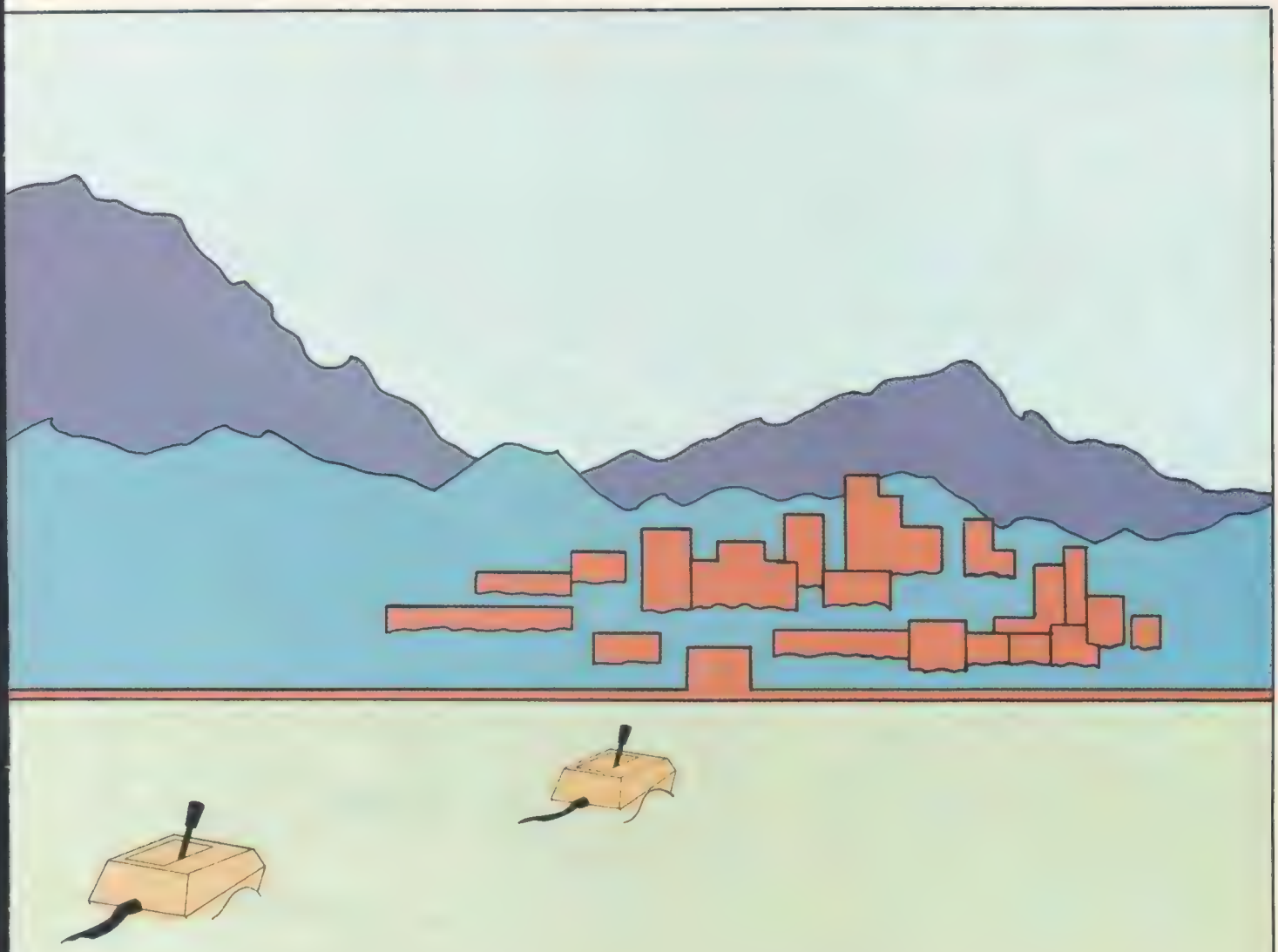
When the Tandy 1000 appeared, I was happy to see two joystick ports. Joystick ports can not only be used for joysticks, but can be used for a variety of other things as well — burglar alarms, measuring wind speed, remote signaling, fire alarms and a host of other applications. All this with very little additional hardware — in some cases only a length of wire and a switch. Sound too good to be true? It isn't — read on.

(William Barden, Jr. is a master communicator in a field in which he is one of the few recognized experts —microcomputers. A prolific author of more than 27 books and handbooks on computers and computer programming, Bill also has authored several instructional software projects for Tandy/Radio Shack.)

Computer Ports — What Are They?

To understand how the joystick port can be used involves a little research into microcomputer *ports* in general and a quick comparison of IBM PC*jr* and Tandy 1000 joystick operation. In the process, we'll reveal what the salesman at your local Radio Shack Computer Center might not tell you — the first release of Tandy 1000 BASIC does not read the joysticks as it should!

Your Tandy 1000 has several types of *interfaces* to the outside world. One interface is the printer port, represented physically by a 34-pin "edge" connector that is used to attach a cable to a system printer. This interface is also called a "Centronics" interface (named for the printer manufacturer) and is a fairly well-defined standard method of



connecting printers to computers, both electronically and physically.

Another interface is the RS-232C interface that is implemented by the Tandy 1000 RS-232C option board (25-1006). Again, this is a well-defined standard for attaching so-called serial devices such as modems or serial line printers to computer systems and terminals.

A third interface is the Tandy 1000 "bus" represented by the connectors on the 1000 "motherboard." This bus is again well defined, in this case by IBM, who first used it on their PC.

Other interfaces are the plug for the keyboard unit, the plug for the RGB monitor and the light pen connector.

Last, but not least, are the two joystick ports. Unlike the printer, RS-232C, and bus interfaces, however, the joystick interface is not a *standard* interface. However, it is much easier to connect to external devices than the other interfaces.

Joystick Operation

I have both an IBM PC*jr* and a Tandy 1000. Tandy modeled the 1000 after the IBM PC*jr* — the high-resolution graphics modes and three channels of sound, in the 1000, for example, imitate these features in the PC*jr*. I, therefore, expected my 1000 joysticks to operate in the same fashion as the joysticks on my IBM PC*jr*. This is not the case.

A typical joystick is shown in Figure 1. The joystick is made up of two variable resistors, one for the X direction and one for the Y direction. The resistors are mechanically linked to the control stick of the unit so that each variable resistor rotates in proportion to the distance moved along the X or Y axis.

6 ROM BANK

plus powerful Nicad battery pack that gives 30 hours of power to your Model 100

$269

A joint PCSG/CRYPTRONICS project.

Now you can access LUCID, DISK+, WRITE ROM, RAM+ and others instantly.

The 6 ROM BANK plus battery pack lets you have the ultimate Model 100 system. When you get your unit just press in up to 6 ROMS into the sockets then push in a simple plug into your Model 100's ROM socket. The ROM bank has two posts on either end that insert into the two little holes on the underside of your Model 100.

The ROM bank props up the Model 100 at the same angle and height as those little legs you've seen. The ROM BANK itself is only about 1½" deep and it runs the width of your Model 100. It only weighs one pound. It not only installs instantly, but it pops free in a second if you need everything to lie flat in a briefcase.

Change from ROM to ROM with the touch of a thumb switch.

You can go from LUCID to WRITE to DISK+ to any other ROMS just by turning the thumb switch at the side of the ROM bank. The 6 ROM BANK is a sturdy well built construction that looks like it is a part of your Model 100.

What is also fantastic is that the ROM bank has a powerful NICAD battery and recharger built right in. This power source supplies up to 30 hours of life to your Model 100 with just 6 hours of recharge. What's nice is that it recharges right from your Model 100's power adapter. This is a quick charge system and if you need power in a hurry, you can get 6 hours of life for your Model 100 by just charging for an hour and half.

Suddenly, the Model 100 is a very powerful computer.

If you have the HOLMES portable disk drive with its powerful bundled database software package or the CRYPTRONICS 128K RAM expansion along with LUCID, WRITE ROM and DISK+ in the 6 ROM bank with its rapid rechargable NICAD power source, you have the ultimate portable system. It's all available only from PCSG.

As usual we don't want you to take our word for it. The 6 ROM BANK is sold on a 30 day trial. If you are not satisfied simply return within thirty days for a full refund. MC, VISA, AM. EX. or C.O.D.

1-214-351-0564

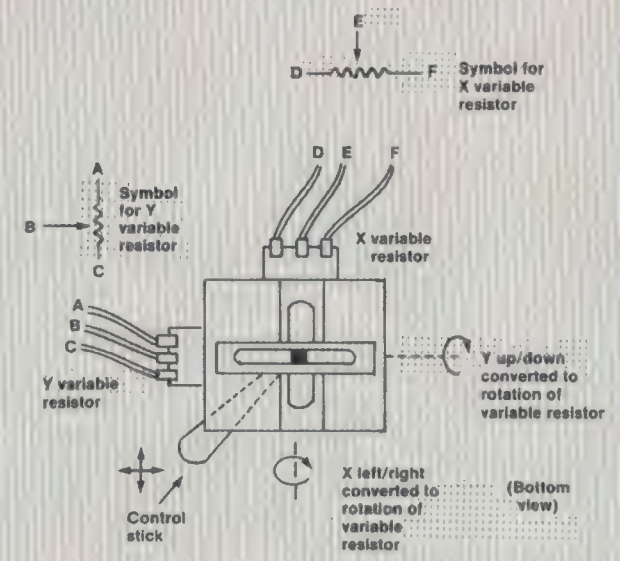
PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

VISA / MASTER CARD / AM. EXP. / C.O.D.

NOW AVAILABLE ON THE M-200

Figure 1. Typical Joystick



Electrical current flow through a variable resistor operates similarly to water flow through a garden hose — squeezing the hose compresses the hose and offers more resistance to the flow of water, diminishing the flow. In the same fashion, a variable resistor offers more or less resistance to the flow of current in an electrical circuit.

The two variable resistors in a PC*jr* joystick are connected to logic within the system that uses the changing resistance of the joysticks, analagous to the physical X and Y positions, to generate a variable width electrical pulse, as shown in Figure 2. The duration of the pulse in the PC*jr* ranges from 24 to 1,124 microseconds (millionths of a second). The two pulses, one for X and one for Y, are read by the PC*jr* BASIC via the STICK command to determine the position. Typical values returned by STICK are shown in Table 1.

Figure 2. Joystick Pulses

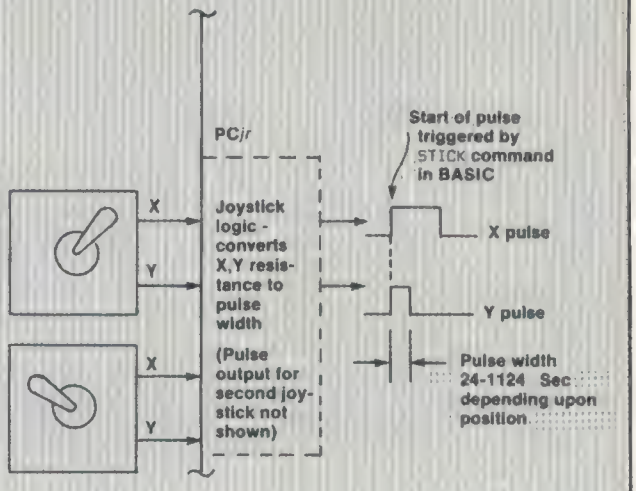


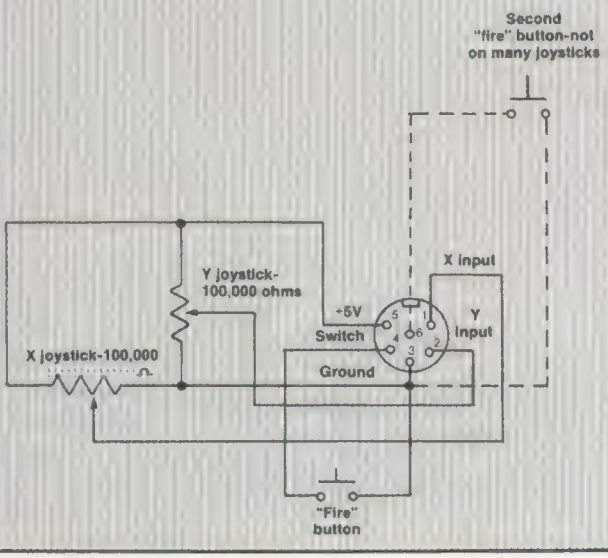
Table 1

PRINT STICK(0) (X)	PRINT STICK(1) (Y)	Position
3	—	extreme left (minimum res.)
60	—	middle
114	—	extreme right
—	3	top (minimum res.)
—	60	middle
—	118	bottom

The values returned by STICK(2) and STICK(3) are for the right joystick and are derived by identical logic.

When I bought joysticks for my Tandy 1000, I noted that the joysticks were identical to those sold for the Color Computer (Radio Shack Deluxe Joystick 26-3012). Physically, these joysticks look very much like the joysticks sold for the PC*jr* — both are made by Kraft (presumably), a company making joysticks for radio control and other applications. However, the PC*jr* and Radio Shack joysticks are not identical electrically. The Radio Shack joystick electrical design is shown in Figure 3.

Figure 3. Radio Shack Joystick Electrical Design



Tandy 1000 STICK Operation

When I plugged in the Deluxe Joystick to my 1000 and tried a simple program to read in the X and Y values, I found strange results. Unlike the PC*jr*, which gave smoothly increasing values as the joystick was moved from left to right and from top to bottom, the 1000 values jumped all over the place. Repeating the operation with a 1000 at the local computer center yielded the same results, and trying a second joystick confirmed the erratic operation — the BASIC STICK command did not work!

One of two possibilities came to mind — either the hardware design was bad or the BASIC interpreter did not read the joysticks properly. One way to prove that the problem was in the BASIC implementation would be to write an assembly language program to read the joystick values independently from BASIC. The resulting program is shown

The listing:

```
10 '****Put this subroutine first*****
12 CLEAR ,25000
14 DATA &h00,&h00,&h8c,&hcb,&h8e,&hdb,&hb4,&h00
15 DATA &hbb,&h00,&h00,&hb9,&h00,&h01,&hba,&h01
16 DATA &h02,&hfa,&hee,&hec,&hd0,&hc8,&h10,&he7
17 DATA &hd0,&hc8,&h10,&he3,&he2,&hf5,&h88,&h3e
18 DATA &ha8,&h61,&h88,&h1e,&ha9,&h61,&hfb,&hcb
19 FOR I=0 TO 39: READ A: POKE 25000+I,A: NEXT I: ZALSUB=25002: GOTO 22
20 CALL ZALSUB: ZX=PEEK(25000): ZY=PEEK(25001)
21 RETURN
22 '*****Exit here first time (not a subroutine)*****
100 '*****Sample Call*****
110 GOSUB 20
120 PRINT ZX,ZY
130 GOTO 110
```

in Listing 1. It is an assembly language program embedded in BASIC that reads the X and Y values of the *right* joystick. To use it, merge the BASIC program in the listing with your BASIC program. RUN 10 only one time to relocate the code. Thereafter, call the subroutine at Line 20 to return ZX and ZY, the joystick X and Y values. With this program, X and Y vary from one to about 91, dependent upon joystick position. A sample call is shown in the listing.

By the way, by the time this article appears there *should* be a new release of BASIC that corrects the joystick problem. I'm including this program as an interim measure just in case the release of BASIC has not shown up by publication. I would expect the new release of BASIC to operate much the same as the PC*jr* version does, returning values of X and Y ranging from about three to a high of about 118.

Detecting Joystick Buttons

The STICK function returns the X and Y values for one or two joysticks, as described above. Another BASIC function, STRIG (and this one *partially* works in the 2.02 release of Tandy 1000 BASIC) detects whether or not the joystick "fire" button has been pushed on the joystick. The fire button, of course, is used for control during games or for enabling or disabling screen functions while using the Radio Shack Color Mouse (26-3025, also compatible with the Tandy 1000, although a higher-priced mouse and adapter board is available). Here's a sample of STRIG:

```
100 STRIG ON
110 IF STRIG(1) THEN PRINT "Button Push!"
120 GOTO 110
```

In the code above, the message "Button Push!" would be displayed during the time that the joystick button is pressed.

There are eight separate "arguments" for STRIG, as follows:

- STRIG(0) Tests whether first button on left joystick was pushed since last STRIG(0) call.
- STRIG(1) Tests whether first button on left joystick is currently being pushed.
- STRIG(2) Same as STRIG(0) for first button of right joystick.
- STRIG(3) Same as STRIG(1) for first button of right joystick.
- STRIG(4) Same as STRIG(0) through STRIG(3) for second joystick buttons.

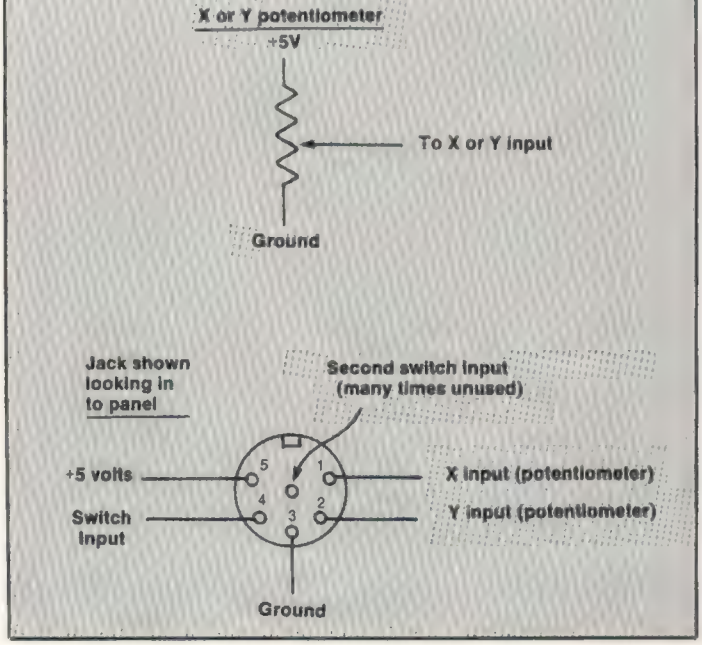
In all cases, a -1 is returned if the button has been pressed and a zero is returned if no press has occurred. As shown in the code above, a STRIG ON should be done before the buttons can be read.

With STICK and STRIG (or with Listing 1 and STRIG), we have all we need to read in the joystick X and Y position and to detect whether or not the joystick button is being pushed from within a BASIC program. Where do we go from here? How can we convert these functions into burglar alarms, windspeed indicating devices, and the like?

More On The Joystick Electronics

It's easy to deduce how the joystick electronics operates. The pin connections for the joystick are shown in Figure 4. The diagram shown indicates the pin positions of the 1000 jack, looking in to the jack on the front panel.

Figure 4. Tandy 1000 Joystick Pin Connections



USE YOUR DESKTOP COMPUTER AS A DISK DRIVE FOR YOUR M-100.

Try *Disk+* for 30 days. If you aren't as excited as we are, return it for a full refund.

When we designed *Disk+* we did it out of necessity. We wanted a way that we could just connect a Model 100 to our desktop computer with a cable and save files onto the desktop's disk drive. We wanted it to be so simple to use it would be self-explanatory.

Picture this. *Disk+* comes to you on a Snap-in ROM and a diskette for your desktop. You take a quarter and open the little compartment on the back of your Model 100. Then you just press the ROM into the socket. *Disk+* appears on your main menu just like a built-in.

You connect your Model 100 to your other computer using an RS232 cable (available from PCSG for $40).

You just place the *Disk+* diskette into the desktop's drive and turn on the computer. It powers up automatically and says "awaiting command" on your desktop's screen. Then you just put the widebar cursor on the Model 100 main menu on *Disk+* and press ENTER. You are shown your RAM files arranged just like the main menu.

To save a file to your other system's disk drive, you just move the widebar cursor to the file you want to save and press ENTER. It is saved instantly with no further action.

To look at the disk directory, you just press a function key on your Model 100. You see immediately the disk directory on your Model 100 screen, and it is arranged just like your Model 100's main menu.

To load a file from the diskette to your Model 100, you just move the widebar cursor to the file and press ENTER. The file is transferred to your Model 100's RAM instantly. You can press F8 and go back to the main menu, and the file you loaded from diskette is there, ready to use.

It is so nice to be able to keep your documents, programs (both BASIC and machine code) and *Lucid* spreadsheet files on the diskette, and bring them back when you need them. All files are ready to run or use with no changes or protocol by you.

If you have access to a desktop computer and don't have *Disk+*, then evidently we have done a poor job telling you about it.

All files and programs that you load or save, go over and come back exactly as they are supposed to be because of full error checking. This guaranteed integrity is really a comfort. *Disk+* is wonderful in so many other ways. For example, you can do a "save all" of all your RAM files with just a touch of a function key. That group of files is saved on the diskette under a single filename with a .SD (for subdirectory) extension. Any time you want, you can bring back all those files at once, or just one or two if you like, again with one-button ease.

Disk+ takes up no RAM. That's zero bytes either for storing the program or for operating overhead.

What really excites most *Disk+* users is text file cross compatibility. Your Model 100's text files are usable on your desktop computer, and your desktop's text files become Model 100 text files.

This means you can write something on your Model 100, and with *Disk+* transfer it instantly to your desktop and start using it right away on your bigger computer. Or the way we like to work is to type in a document on the desktop computer and then transfer it to our Model 100 with *Disk+*. Then we print out the document, beautifully formatted, using WRITE ROM.

Disk+ works with just about every micro sold, from IBM PC and its clones, to all Radio Shack computers (yes, all), to Apple II, Kaypro, Epson and most CPM. Just ask us. More than likely, your computer is supported.

Incidentally, hundreds of Model 100 owners have gone to their Radio Shack stores and bought a color computer because it is so low priced, and with *Disk+* they have an inexpensive disk drive.

And if that weren't enough, how about this: *Disk+* also provides cross-compatibility between different computers like IBM, Apple or the Model 4 using the Model 100 as the intermediary device. Quite a feature!

The snap-in ROM is really great because you can use other ROMs like *Lucid* or WRITE ROM. They snap in and out as easily as an Atari game cartridge and you never lose your files in RAM.

Anyone who ever uses *Disk+* simply can't do without it. But so many times we have had new users call us and say, "Wow! I had no idea when I ordered it that *Disk+* would be so fantastic. I just couldn't believe that I could use my desktop computer's disk drive with my Model 100 just like it is another main menu."

That's why we sell *Disk+* on a thirty-day trial. If you aren't completely satisfied, return it within thirty days for a full refund. Priced at $149.95 on Snap-in ROM. MasterCard, Visa or COD.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

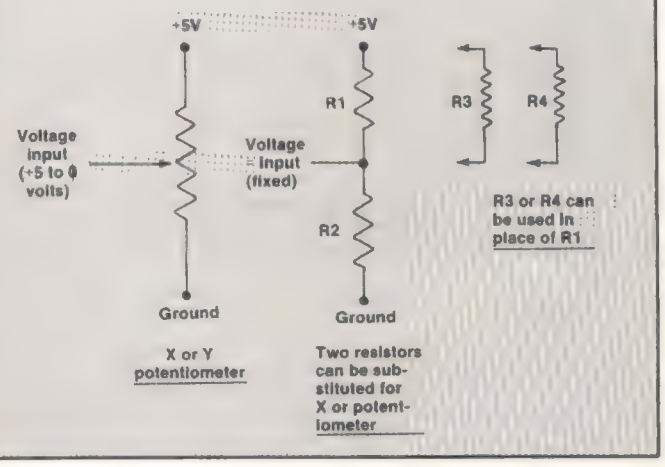
11035 Harry Hines Blvd., #207, Dallas, Texas 75229 • 1(214) 351-0564

PCSG provides hotline software support for the Model 100. Call us at 1-214-351-0564

The variable resistor (called a "potentiometer") acts as a "voltage divider" — one end is attached to a voltage of +5 volts, a common microcomputer voltage, and the other end is "ground" or zero volts. The potentiometer "wiper" rides between these two voltage extremes. As the wiper moves from the +5 volt end down to the zero volts end, voltage output on Pin X falls from +5 volts to zero volts. Using the garden hose analogy again, it's as if the hose was 1,000 feet long with just a trickle of water coming out of the nozzle end with full pressure at the faucet end. Poking a hole in the hose near the faucet end would produce a higher pressure leak than a hole near the nozzle end. A hole midway along the hose would produce an intermediate pressure, and so forth.

The secret to substituting external devices for the joystick is that two simple "resistors" can be used in place of the X or Y potentiometer, as shown in Figure 5. The output at the junction point is always a fixed voltage when this is done. However, additional resistors can be added to change the voltage to other levels, as shown in the figure. Switches control which resistors are connected and when.

Figure 5. Using External Resistors in Place of X or Y Potentiomenter



The switch inputs on pins 4 and 6 are normally "floating." When a button is pushed, the output is connected to ground, or zero volts. The switches may also be external switches, located remotely from the 1000.

Wires in speaker or intercom cable (such as Radio Shack Speaker Wire 278-1385 or Intercom Cable 278-371) has very little resistance compared to the resistance of the potentiometers. The resistance of the potentiometers is 100,000 ohms at full scale, while the resistance of 1,000 feet of intercom or speaker wire is typically 26 ohms. The remote resistors or switches, then, may be located hundreds of feet away and read by the `STICK` and `STRIG` commands.

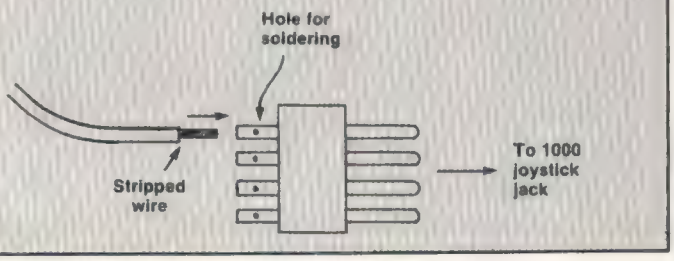
If you feel that electronics is beyond your ken, don't despair. We'll make the following applications foolproof. Another worry you might have are the mysterious problems connected with voltage and current. The voltage used on the joysticks is extremely low (+5 volts), and you can treat it with impunity. You won't "burn anything out" inside the 1000, either. While it's true that enough current from +5 volts can arc weld screwdrivers together, the joystick voltage is "current limited." However, keep the +5-volt lead from the joystick port away from the ground lead. We found the system reset when the two were inadvertantly shorted.

A Standard Joystick Plug

Tandy doesn't want you fooling around inside their 1000s and connecting "foreign" devices to them. Contrarily, however, they do sell a joystick plug that will enable you to get at the joystick port. It's the Radio Shack DIN Plug 274-020, a 6-pin connector. In addition to the plug, you'll need half a dozen clip leads (Radio Shack Test Lead Cables 278-1156) and an assortment of resistors (Radio Shack ¼-Watt Resistor Kit 271-308). The clip leads and resistors are not at all critical. Any small clip leads will do. For resistors, use ¼ watt or greater (½ watt, etc.) and use values close to those indicated in the text. Also required for some applications are switches. Again, these are not at all critical — you can use anything from a switch designed for a Hoover Dam substation to a micro-miniature switch. Buy the ones designated "SPST" or "SPDT" (single-pull, single throw or single-pull double-throw) such as Radio Shack Mini SPST Momentary (275-1547) or Submini SPST Momentary (275-1571). Also required for remote applications is multi-conductor cable. I'd recommend Radio Shack Intercom Cable 278-371 — it's stranded instead of solid and hence will not break as easily. It also comes with three wires per cable, allowing you to hook up two inputs. For more conductors, use Radio Shack Six-Conductor Telephone Cable 278-374.

Unfortunately, the joystick jacks are recessed inside the front panel and it's difficult to use test leads clipped on to the exposed pins. A better solution is to solder the wires that you need onto the exposed pins on the back of the plug. Figure 6 shows the procedure. Each pin has a small hole through which solder can flow. After soldering, look for solder "bridges" between adjacent pins. They are to be avoided.

Figure 6. Preparing the Joystick Plug



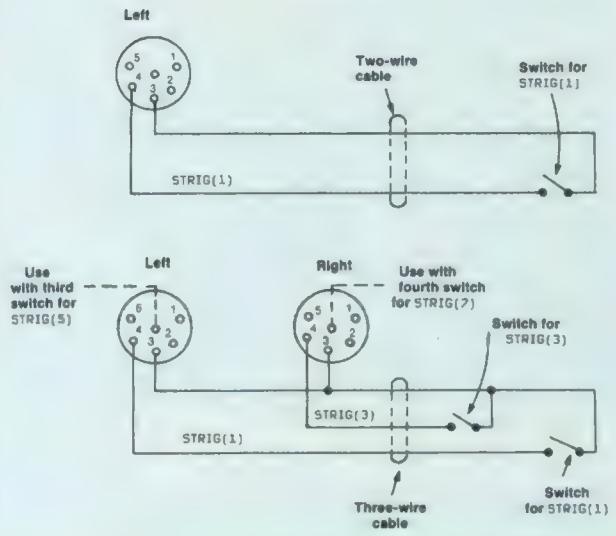
WARNING

Never connect the joystick port wires to any other source of voltage, and never, *ever* connect to power line (AC) wires. Power line voltages can easily kill. At the very least you may zap a Tandy 1000 component or many components, resulting in an astronomical repair bill. The applications to follow are perfectly safe, but use the specified components only, without additional power sources.

Remote Switches

For a single switch operated from hundreds of feet away, use the design shown in Figure 7. The switch closure can be read by a `STRIG(1)` command. For two switches, use an additional switch on a second plug as shown in the figure. For three or four switches, use the second switch input(s). The switches may be those for burglar alarms (Radio Shack Door/Window Switch) or for other functions.

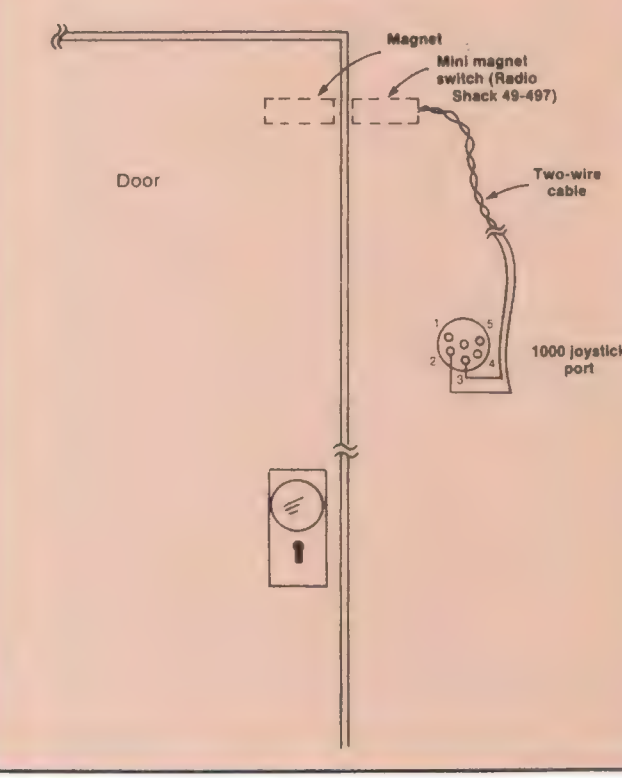
Figure 7. Remote Switch Application



As an example of an application of this type, you might want to record all of the times that someone opened a door to your room. The setup in Figure 8 uses a magnetic switch that closes for each door entry. The program shown below records the times:

```
100 STRIG ON
110 IF STRIG(1)=0 THEN 110
120 LPRINT "Door opened at "; TIME$
130 FOR I=1 TO 10000: NEXT I
140 GOTO 110
```

Figure 8. Door Opening Detector



The delay loop in Line 130 delays approximately 15 seconds to avoid repetition in printing.

Remote Switches That Interrupt A Program

BASIC has a built-in interrupt capability for the joystick switches. This means that pressing a joystick button will trigger a suspension of the BASIC code being executed and a jump will be made to a special interrupt processing subroutine. Here's an example of how to set up the interrupt:

```
100 ON STRIG(0) GOSUB 10000
110 PRINT "Main Code"
120 GOTO 110
.
. (additional program code)
.
10000 ' Interrupt Processing Subroutine
10010 PRINT "Interrupt Processing Entered"
10020 FOR I=1 TO 1000: NEXT I
10030 RETURN
```

If you run this short program, you'll see the message "Interrupt Processing Subroutine" displayed whenever the left joystick button is pushed — the joystick button has caused an interrupt of the the "main" program. At the end of the interrupt action, the main code is re-entered. This example can be expanded as much as required, and you could take any action necessary in the interrupt subroutine. You could also have up to four separate interrupt subroutines, one for STRIG(0), (2), (4) and (6). (0, left joystick Button 1; 2, right joystick Button 1; 4, left joystick Button 2; 6, right joystick Button 2.) However, the interrupt capability works only for a BASIC program running as a main program with an interrupt subroutine also resident in memory. You cannot use it with, say, *Lotus 1-2-3*, and get the interrupt action.

Multiple Remote Switches

A slightly different design from the first implements many separate remote switches, each with their unique "code," as shown in Figure 9. Each switch is at a separate "station," a station being a burglar alarm sensor, fire sensor or other application. In the example shown, eight separate stations have been used. Each station switches a unique resistance value which is read in by the STICK function. A comparison is then made to see which of eight ranges is active. The BASIC code looks like this:

```
100 '*****Eight-Station Decoder*****
105 GOSUB 20: VALUE=ZX
110 IF VALUE<7 THEN 105
120 IF VALUE<20 THEN A=1: GOTO 200
130 IF VALUE<28 THEN A=2: GOTO 200
140 IF VALUE<37 THEN A=3: GOTO 200
150 IF VALUE<45 THEN A=4: GOTO 200
155 IF VALUE<56 THEN A=5: GOTO 200
160 IF VALUE<65 THEN A=6: GOTO 200
170 IF VALUE<75 THEN A=7: GOTO 200
180 A=8
200 PRINT "station#:";A;"active"
210 FOR I=1 TO 350: NEXT I
220 GOTO 105
```

IF YOU'RE MISSING THIS,

Add Multiplan or Interactive Solutions to

If you own a Model 100, then you already know. You know that the Model 100 comes with five built-in management programs. You know about the full-size typewriter-style keyboard and the direct-connect modem and telephone auto-dialer. But did you know how easy it is to boost your portable's power with Multiplan or Interactive Solutions?

You're really missing out if you don't know how easy it is to supercharge your Model 100. Just turn your machine over, pop in either of these high-performance ROM modules, and you've got a powerful little machine that can do anything from complex spreadsheet analysis and calculations to sales forecasts, profit-and-loss projections, budgeting, pricing, engineering calculations and much more.

Powerful Spreadsheet

Let Multiplan (26-3929, $149.95) replace your pen, paper and calculator. Multiplan is an electronic worksheet—a large grid for entries, each of which can be words, titles, numbers or formulas. Multiplan then performs complex calculations ... instantly! And because it remembers relationships between entries on a worksheet, Multiplan automatically performs calculations with a large number of variables. This lets you test plans by putting different values into your formulas.

You can run sensitivity analysis, do budget and resource planning, and schedule more efficiently. You'll soon find that Multiplan is a vast improvement over "hand calculating." Multiplan overcomes the limitations of paperwork. It offers a worksheet with up to 99 rows and 63 columns. You can instantly insert or erase data and widen or shrink columns, eliminating the costly and time-consuming work of typing or hand-printing the worksheet over and over. A Multiplan worksheet is always flexible.

YOU'RE MISSING OUT!

Supercharge your Model 100 Portable.

Get Three Programs for the Price of One

Interactive Solutions (26-3844, $149.95) is a three-in-one cross-referencing software package with database management, spreadsheet analysis, and word processing.

Because Interactive Solutions is a ROM add-on, it increases the amount of memory that can be utilized by your Model 100. In addition, you can save data on cassette instead of using valuable RAM memory.

Information stored in Data Manager, the data base, can be utilized by Data Calc, the Word Processor allows you to merge Data Manager information into your text. Or you can select any part of a Data Calc worksheet and add it to any text file in Word Processor.

Interactive Solutions is easy to learn too, because it's designed to be self-explanatory and guide you through each process. Clear and concise labels and menus make decision making easy.

Come in Today!

Supercharge your Model 100 today. Find out how at your nearest Radio Shack Computer Center. Hurry in and see what you've been missing.

Radio Shack®
The Technology Store™
A DIVISION OF TANDY CORPORATION

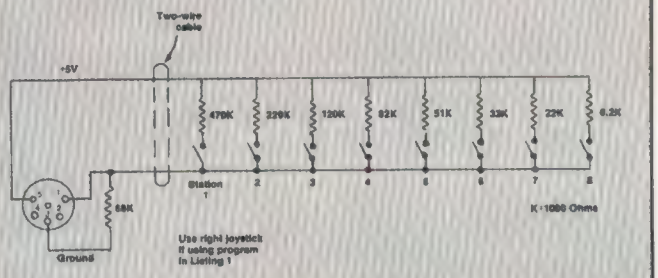
Send me a free Computer Catalog.

Radio Shack, Dept. 86-A-915
300 One Tandy Center
Fort Worth, TX 76102

Name _____
Company _____
_____
Address _____
City _____
State _____ Zip _____
Phone _____

Prices apply at Radio Shack Computer Centers and at stores and dealers. Multiplan/TM Microsoft

Figure 9. Multiple Remote Switch Application



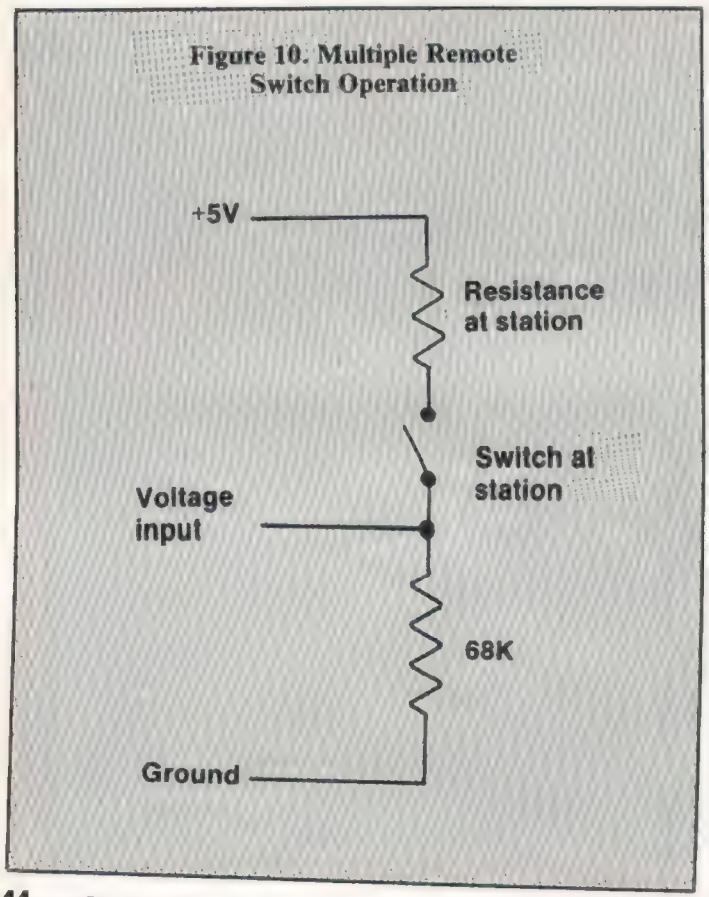
This code compares the value input and finds which station is active, if any. A delay loop prevents repetitious printing of the "Station X Active" message if the button is held down less than about half a second. The values given are for the joystick program of Listing 1. Use appropriately larger values for the new version of BASIC, if available.

To construct the remote stations use a single switch and a single resistor for each station, as shown in the figure. The switch/resistor combination is attached across a common pair of wires that lead back to the joystick port. At the joystick port, there are three wires connected, along with a final resistor.

When a switch is pushed at a remote station, the resulting circuit is shown in Figure 10. The input voltage (and resulting delay time and STICK value) is given by the following formula:

$$\text{Voltage} = 5 \text{ volts} \times \frac{68{,}000}{\text{Resistance at Station} + 68{,}000}$$

Figure 10. Multiple Remote Switch Operation



As each station uses a different resistance value, eight unique values are read by the STICK(2) function. The nominal values read (for the subroutine shown in Listing 1) are:

Station #	Resistance	Value
1	470,000	15
2	220,000	23
3	120,000	33
4	82,000	41
5	51,000	52
6	33,000	61
7	20,000	70
8	8,200	81

Some experimentation might be necessary to determine proper resistance values for your system.

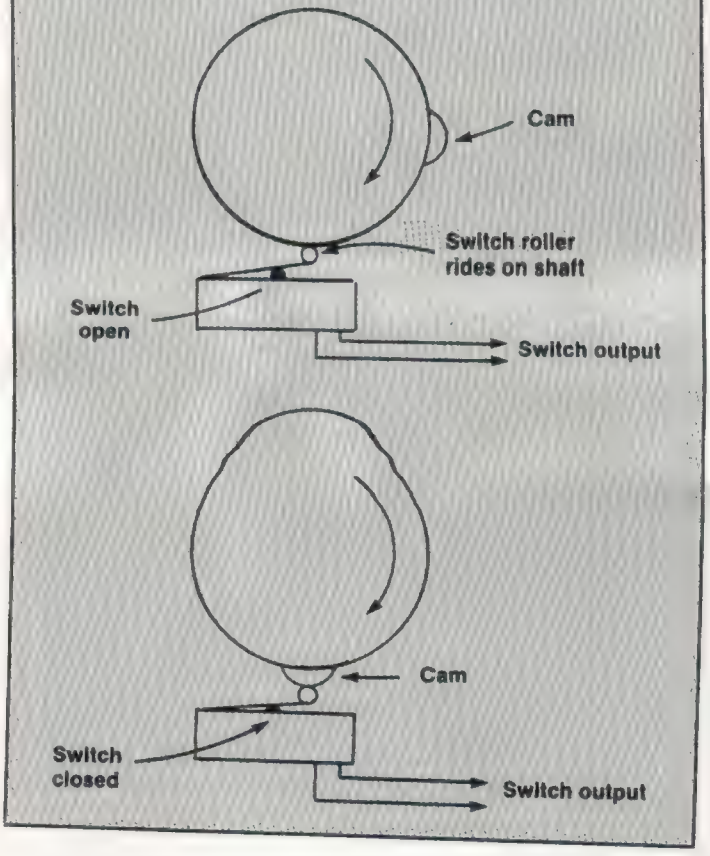
This concept can be used with a larger number of stations than eight. However, if too many stations are used, there may be a problem in *resolution* between one station and another as the resistance values approach one another. Up to 32 stations can be used with no difficulties if all four STICK inputs (0), (1), (2) and (3) are used.

Rotation Sensing

The idea of a remote switch can easily be used for measuring the rotational speed of shaft. For this purpose, two type of switches are useful, a "microswitch" and a "magnetic reed switch."

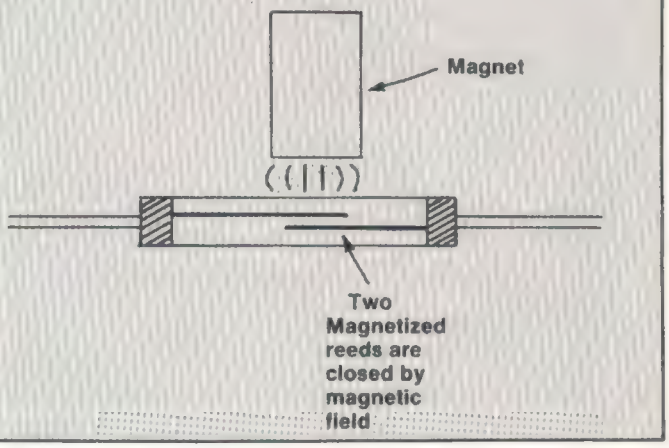
The microswitch is simply a switch that takes very little force to operate. Radio Shack carries one (Sub-Mini Lever Switch 275-017) that takes a force of only 5 grams to operate, approximately equal to the weight of five quarters. The microswitch has a roller on the actuating lever that can follow a shaft as it rotates. A cam or bump on the shaft will close the switch, as shown in Figure 11.

Figure 11. Microswitch on A Camshaft



The magnetic reed switch contains two contacts that are closed by an external magnet, as shown in Figure 12. A typical distance between the reed switch and magnet is ⅛ of an inch. The magnetic reed switch, therefore, requires no physical contact with a rotating shaft for operation. Radio Shack carried reed switches at one time, but no longer have them in their catalog. However, they are commonly available from other electronic parts suppliers, and they are inexpensive.

Figure 12. Magnetic Reed Switch



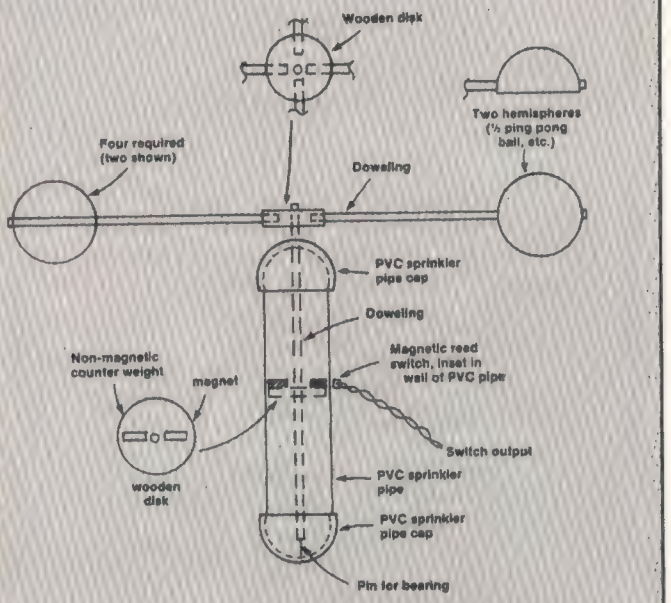
As an example of what can be done with a magnetic reed switch, see Figure 13. An anemometer is shown, constructed out of commonly available materials. A magnet is mounted on the shaft of the anemometer and triggers a magnetic reed switch. The switch contacts go to the STRIG input of the joystick port and can be sensed by the STRIG function. Since the STRIG function returns a "true" condition as long as the switch is on, some logic must be incorporated into the program so that only one switch closure is read per rotation. Here is a sample:

```
100'*****Anemometer Routine - Prints RPM*****
110 COUNT=0
120 TIMER ON
130 ON TIMER(60*18.2) GOSUB 170
140 STRIG(0) ON
150 COUNT=COUNT+STRIG(0)
160 GOTO 150
170 PRINT "Revolutions per minute=";ABS(COUNT):
    COUNT=0
180 RETURN 130
```

This code works as follows: The STRIG(0) function is used to detect if a switch closure occurred since the last STRIG(0) function, rather than detecting the instantaneous condition of the switch. The STRIG(0) command automatically sets up BASIC to detect this condition. As long as switch closures do not occur faster than the BASIC loop from Line 150 through Line 160, every separate switch closure will be counted.

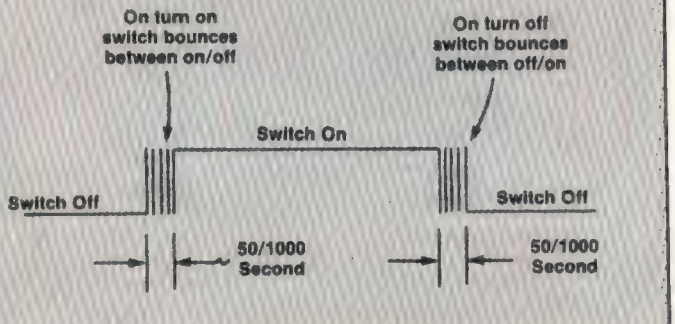
The ON TIMER statement sets up an automatic interrupt at the end of 60 seconds. (The value in the ON TIMER statement is in "clock ticks" and not seconds as the Tandy 1000 BASIC manual states; this is a possible BASIC error, as the PC*jr* uses seconds.) At this time, the subroutine at Line 170 is entered. The code in the subroutine prints the count and then returns to another one-minute sample.

Figure 13. Simple Anemometer



Some mention must be made here for the necessity of "debouncing" switches. In previous code, we put in a delay loop when reading the instantaneous state of an external switch. When switches operate, minute "makes" and "breaks" occur over small periods of time before the switch contact is firmly made or broken, as shown in Figure 14. We experienced no difficulties with switch debouncing in this code at low revolutions; if, however, you see spurious counts, you will have to put in a short delay loop to bypass the time during which switch bounce occurs. Evidently there is enough "overhead" in this loop to handle the switch bounce condition.

Figure 14. Switch Bouncing



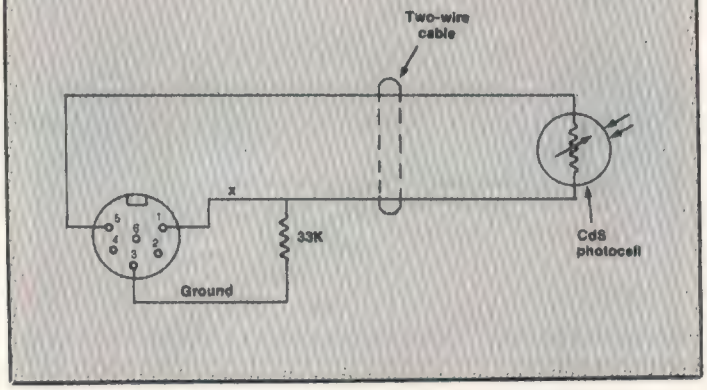
Interpretive BASIC is fast enough to keep pace with a moderately rotating shaft. This method is good for shaft rotations of about 200 revolutions per second (12,000 RPM). For the anemometer application discussed above, that's equivalent to windspeeds of about 350 miles per hour!

A Light/Dark Detector

Our final application measures ambient light or can be used to detect a dark or light condition. Some possible applications are a detection of when a mailbox is opened; detection of when the lights in a room are turned on; or a device to measure cloudy day conditions. Believe it or not, all of this can be done for about $1.50 (and $1,000 worth of Tandy 1000). The design is shown in Figure 15. The design uses a cadmium sulfide photocell and a single resistor. The cadmium sulfide photocell is really just a variable resistance. Here are some resistance values of the photocell for various conditions:

Condition	Resistance	STICK Value
Black hole	infinity ohms	0
Interior room, very dim light	23,000	27
Interior room, sunlight, facing in	3,500	67
Interior room, sunlight, facing out	1,600	78
Outside, cloudy	200	89
Outside, sunlight	65	90

Figure 15. Cadmium Sulfide Photocell Application



As you can see, the most useful range will be from very dim room light to the typical light found during a sunny day in a well-lighted room. The values shown are for the program of Listing 1. As with the other applications, the CdS device can be located hundreds of feet away from the system.

Blue Skying It

The above applications are really just a hint of what can be done in using the joystick inputs. There are many more real-world applications that could be done, and very simply — things such as measurement of temperature, humidity, rainfall and moisture, audio detecting devices and digitization of images.

The beauty of using the joystick port is that the logic is already built in, and very little external hardware is required. Look into some of these applications yourself, but don't tell Tandy I suggested using the joystick port for them! PCM

Changes your Model 100 into a totally different computer with capability you never thought possible.

Lucid™ on Snap-In™ ROM

Infoworld rated Lucid's performance "excellent"

NOW AVAILABLE ON THE M-200

PCSG says "Satisfaction Guaranteed or your money back within 30 days!"

LUCID© is here now. It is on a ROM cartridge that snaps into the compartment on the back of your Model 100. It takes no memory to load and no memory for operating overhead. That means you have the full 29.6k bytes free to store your data.

First, LUCID© is memory conserving. It will let you build a large spreadsheet – 255 row by 126 column capacity. You build huge spreadsheets in your Model 100's RAM that could consume 80 to 100K on a desktop computer.

Secondly, LUCID© is fast. LUCID© is so rapid, a 36 column corporate financial statement took less than 4 seconds to calculate.

Thirdly, LUCID© has features you won't find in most other spreadsheets. For example, when you type a label (text) it will cross column boundaries; in other words when you type a label or title it will appear as you type it irrespective of column or width. LUCID© also allows you to set column widths individually, and of course LUCID© has insert row and insert columns, as well as other standard features. LUCID© even lets your formulas refer to cells in other spreadsheet files.

Further, LUCID© has what no other spreadsheet has: Cut, Copy, and Paste. It uses the same keys as Cut and Paste in TEXT, but here's the difference: it takes all the formulas with it when you paste and they all automatically recalculate with the entire sheet.

And here is what is really amazing. You can copy or cut from one spreadsheet and paste into another spreadsheet or even a TEXT file.

LUCID©supports all BASIC math functions as well as Log, sine, cosine, tangent, exponentiation and other sophisticated math functions.

LUCID© has so many features that you will say "this is what I need in a spreadsheet", such as automatic prompting of an incorrectly typed-in formula showing just where the mistake was made.

LUCID© has expanded "go to" functions that remember and produce a windowing capability.

But perhaps most remarkable is that LUCID© is not only a spreadsheet but a program generator as well. First, LUCID© lets you protect all cells against entry or change, and then unprotect just the cells you want for someone else to use as input fields.

LUCID© will not only process values, but text input as well so that the facts other than numbers can be responded to. LUCID© has the ability for you to refer in a formula to cells containing words. This feature combines with the capacity of doing "if then" statements that work by doing table lookups against even massive X/Y charts of text or numerical information. You can produce a program that responds to inputs with no programming knowledge whatsoever.

You can prepare a report section in your spreadsheet with instructions to your user for printout, and they can produce a personalized printout that responds to their input. All your formulas and tables that did the calculations and provided the facts are invisible to that user. LUCID© is useful for doctors for patient questionnaires, troubleshooting technicians, purchase clerks, people doing job quotes, stores for customer workups, insurance agents and anybody who needs to process specific facts and numbers to produce a report based on those responses.

LUCID© comes with a manual that explains not only the characteristics of LUCID©, but will train you how to use a spreadsheet even if you have never seen one before. You are shown how to do budgets, forecasts, breakeven analysis amortizations and many other types of personal and business reports and calculations.

User friendly is such an over-used term in this industry, but a typical comment has been "I have never seen a spreadsheet that does so much, and yet LUCID© is so much easier and faster to use."

LUCID© is a result of a most exhaustive developmental effort in which PCSG's objective was to develop a spreadsheet that was better than the state-of-the-art. We are so pleased because LUCID© provides for the Model 100 spreadsheet capability you cannot equal on a desktop computer.

LUCID© is, in our opinion and that of those who have examined it, a breakthrough. We sell it on a 30 day trial. If you are not completely satisfied, return it within 30 days for a full refund. Priced at $99, on snap-in ROM. MasterCard, Visa or COD.

First with software for the Model 100

PORTABLE COMPUTER SUPPORT GROUP

1-214-351-0564

11035 Harry Hines Blvd., #207
Dallas, Texas 75229

Add textures and patterns to color graphics displays

Fun with Filters

By H.W. Kemp

In this article we will use some very simple BASIC programs to create color and texture changes to Tandy 2000 graphics screens. You will need a diskette with about 100K free in Drive B. Your MS-DOS disk in Drive A should be a backup copy which you intend to use just for graphics. It should have about 6K free.

We will need a demonstration graphics display to add filters to. To keep things simple, let's use CIRCLE.BAS from The Gallery (PCM October 1985). If you have already keyed it in, you are ready to proceed. I have included the program as Listing 1. Note two changes: I have added 20 to the first X dimension of the CIRCLE commands to center the pattern on the screen and I have deleted Line 1340.

I have added Line 1015 so that you may see how the pattern would look in low resolution by changing Line 1000 to read MODEL=2001. Run the program at least once this way and feel real smug about your Tandy 2000's great screen resolution!

The program should be saved as B:CIRCLE.BAS.

Now I want to introduce you to a very simple procedure we will be using throughout this little exercise.

(H.W. Kemp lives in Gainesville, Florida where he uses his Tandy 2000 system for programming and word processing. He has been the featured artist in "The Gallery.")

We'll want to create a screen dump so we will not have to wait for the BASIC program to create a graphics screen for us each time we run the program. Listing 2 is the program which does this (from the Tandy 2000 manual). Type NEW and key in the program. Now save this code in ASCII format to Drive A (SAVE"SCRDISK",A).

Our next step will be to load B:CIRCLE and add SCRDISK to the program with the MERGE command: MERGE"SCRDISK". (Starting SCRDISK with a high line number eliminates problems of overwriting the existing program when merging.)

Now run the combined program. It will be saved to disk automatically. After the disk drives have stopped running, type NEW

Refer to Listing 3. This is the program furnished by Tandy which will bring the screen back from the disk. Let's call it DISKSCR. Type this in and save it in ASCII format to Disk A: SAVE"DISKSCR",A.

If you haven't used this screen saving and recalling procedure before, type RUN and enjoy. The image will come up in three color screens very rapidly.

Our next step will be to create the "filters." We will save these in ASCII format to Disk A. By the end of this session, we will have a library of these overlays, which we can merge with our DISKSCR program to add color and textures to the original.

Let's start with a very simple "filter," vertical lines added to the screen. (Listing 4.) It is important to start lines with a large number to eliminate problems when merging. I have used 60000. Also, it is desirable to include the remark statements. This lets you identify the particular filter after you have merged several to the same screen.

Type NEW and key in the listing of FVERTICL. Save it to Disk A in ASCII format: SAVE"FVERTICL",A.

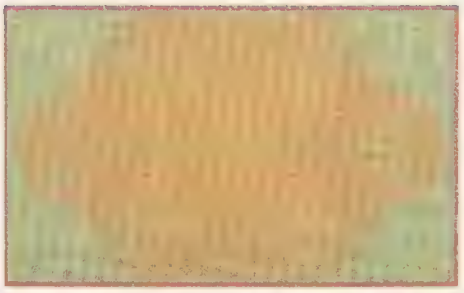
We now have a simple overlay which will add a color on top of our screen. Let's try it. Type LOAD"DISKSCR" and press ENTER. Now type MERGE"FVERTICL" to merge the two programs. Run it. The circle pattern will be dumped to the screen and cyan-colored vertical lines will be added to it.

If another color is desired, all we have to do is change Line 60030 to LINE(X,Y1)-(X,Y2),1. The last number is the color designator; if we change the 1 to 4, we should get red lines. List the program, change 1 to 4 in Line 60030 and run it. Pretty neat!

We can also adjust the line spacing by changing the STEP statement in Line 60020. List the program, change the 3 to 5 in 60020 and run it. All of the programs in this article can be modified in much the same manner.

Our next filter will be FBORDER. (Listing 5.) We will use simple line commands to create a neat border for our masterpiece. Type NEW and key in this listing. Notice that I have used the box command to create seven boxes —

The original CIRCLE.BAS program with cyan colored lines added and a border.



The original CIRCLE.BAS program with both vertical and horizontal lines added with a border. (The color statement has been changed to 7 - yellow.)

The original CIRCLE.BAS program with vertical and horizontal lines added with a border. Background color has been changed to blue. The LINE command color statement has been changed to 2 - magenta.

The original CIRCLE.BAS program with a grid of dots added and a border. The color statement of the PSET command has been changed to 2 - magenta.

The original CIRCLE.BAS program with expanding circles (beginning at the center) and a border. The color statement in the CIRCLE command has been changed to 2 - magenta.

The original program CIRCLE.BAS, with expanding circles and vertical lines added and a border. The color statement in the CIRCLE command has been changed to 2 - magenta. The color statement in the LINE command was left at 1 - cyan.

six red and one blue. Don't forget to save it in ASCII format: SAVE"FBORDER",A.

Now enter LOAD"DISKSCR", MERGE"FVERTICL" and, before we enter MERGE"FBORDER", list the program. We have a problem. If we merge FBORDER, it will replace FVERTICL. The solution is to use the RENUM command before we merge FBORDER.

The series of commands is LOAD"DISKSCR, MERGE"FVERTICL", RENUM and MERGE"FBORDER". Now when we run the program, we get a nice cyan-colored vertical line repeating across the screen and a neat border.

Let's review the procedure which you may use with the rest of the listings. *Load* the program that brings up the screen dump, *merge* in one of the filters then *renumber* before *merging* another filter. You can add as many as you like, but remember that you *must* use the RENUM command between merges.

Of course, we could have added these filters to the original BASIC program, CIRCLE.BAS, just as easily, but I find it more useful to add them to a screen dump because I am often just trying out ideas and this method is much faster.

A list of the additional filters follows with a brief description of each. Key them in and try them.

- FGRIDLNE produces a square grid over the screen with vertical and horizontal lines.
- FHORIZN makes horizontal lines.
- FGRIDOTS uses the PSET command to create a grid of dots which produces a very soft color change.
- FOVALDOT is an oval pattern made up of dots centered on the screen.
- FCIRCLE covers the screen with circles expanding from the center of the screen. This is particularly effective with our demonstration screen. For a dramatic pattern, add FVERTICL to this.
- FCOSINE is a rather wild frame made up of four colors.
- FDIAGLFT, FDIAGLDN, FDIAGLUP and FDIAGLER2 are steps leading to FDIAGLR4, which draws diagonal lines across the screen from the four corners. I suggest you key in FDIAGLR4 at first. Perhaps you won't want the others.
- DIAGPLS adds verticals to the diagonal group above, creating rather spectacular results. *Don't key this in.* Just type NEW, LOAD"FDIAGLR4", RENUM, MERGE"FVERTICL" and type RENUM again.

If you have difficulty understanding the programs, review LINE, CIRCLE and PSET in your Tandy 2000 manual.

Note that the format in each case involves a FOR/NEXT loop with a STEP statement to move the element across the screen. The element is developed using LINE, PSET and CIRCLE commands. The length (or radius) of the element is also defined.

Experiment on your own. Change spacing of elements by changing the STEP statement. Limit the extent of the screen covered by changing the second number in the FOR statement — FOR X=0 TO 640 could become FOR X=0 TO 320, thus covering just half the screen. Color half the screen one color, change for the other half. Alternate colors — for example, in FVERTICL, LOAD"FVERTICL", RENUM and MERGE"FVERTICL". Then change Line 60020 to FOR X=1 TO 640 STEP 3 and change the color statement of this second listing. The result will be a "candy stripe" across the screen. The possibilities are limited only by your imagination.

Try printing the filters alone. You will get an error message when you run a filter unless you remember to type SCREEN 3. One spectacular combination is LOAD"FCIRCLE", RENUM, MERGE"FVERTICL", RENUM and MERGE"FBORDER".

I'm certain that there are much more elegant ways to add color and texture to Tandy 2000 graphics, but I have been delighted to find that very simple BASIC statements can add some additional graphics tools to enhance the output of this wonderful machine. Enjoy! □

The New Tandy® 3000

The difference is power ...and affordability.

Introducing the Tandy 3000, the affordable alternative to the IBM® PC/AT. Here's the power you need to manage your business, to network computers, or to create a multiuser system.

Unmatched compatibility

The Tandy 3000 uses the advanced MS-DOS 3.1 operating system. And since the Tandy 3000 is compatible with programs designed for the PC/AT, as well as the PC/XT, it cuts through today's software confusion. Choose from advanced word-processing packages, database management and powerful accounting software.

Power to share

The Tandy 3000 is also designed to use the forthcoming XENIX 5.0 multiuser operating system. Two to six people in an office can use the 3000 simultaneously with low-cost data display terminals.

In offices already equipped with MS-DOS computers, the Tandy 3000 is the link that brings them all together. Using our ViaNet local area network, the Tandy 3000's high speed is available to all network users. The Tandy 3000 is your affordable solution for office automation.

Designed for High Performance

The Tandy 3000 comes with an 80286 microprocessor for the ultimate in speed and performance. Its 16-bit architecture operates at 8 megahertz—twice that of the industry standard.

The Tandy 3000 comes with one-half megabyte of main memory (512K RAM). This memory can be expanded to 640K without using an expansion slot. Total internal memory can exceed an astounding 12 megabytes (under XENIX).

As for storage, the Tandy 3000 (25-4001, $2599) is equipped with a high-capacity, 5¼" slim-line floppy disk drive. For the utmost in compatibility, this drive can read 1.2-meg and 360K formats for use with any IBM PC diskettes.

For maximum storage capacity, choose the Tandy 3000 HD (25-4010, $3599) with a built-in 20-megabyte hard disk drive for fast access to volumes of data.

Both versions of the Tandy 3000 feature a built-in real-time clock for time-sensitive applications and a serial/parallel interface.

The Tandy 3000 has ten expansion slots, including seven IBM PC/AT-compatible slots, two PC/XT-compatible slots and a PC/XT-compatible half-slot for the serial/parallel interface board.

Tandy...Clearly Superior™

High performance, compatibility, multiuser and networking capabilities, and remarkable expandability: the Tandy 3000 has the power to put you in command.

Available at over 1200 Radio Shack Computer Centers and at participating Radio Shack stores and dealers.

Radio Shack®
COMPUTER CENTERS
A DIVISION OF TANDY CORPORATION

Send me a Tandy 3000 brochure.

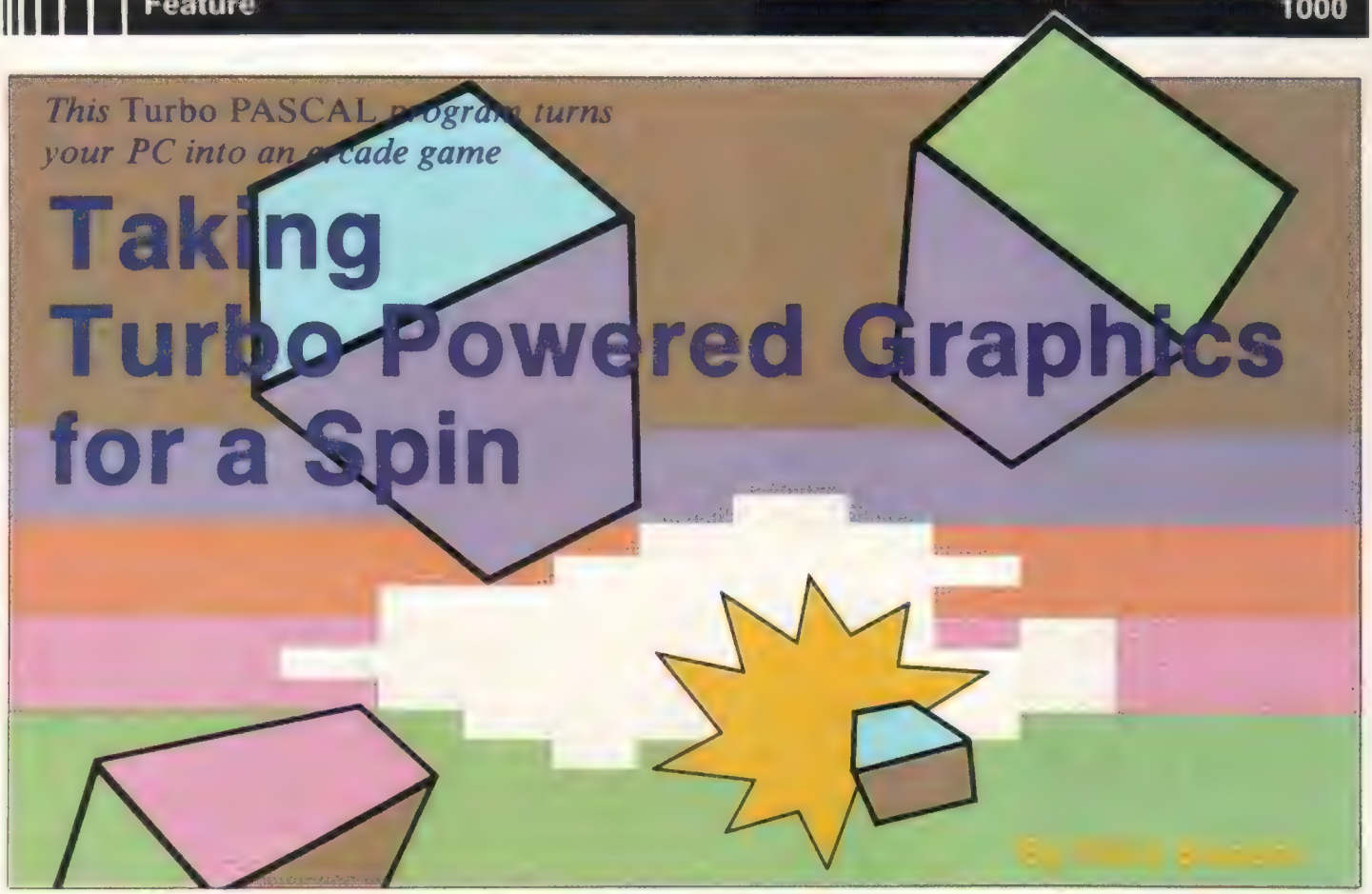
Dept. 86-A-989, 300 One Tandy Center
Fort Worth, Texas 76102

Name _____
Company _____
Address _____
City _____
State _____ Zip _____
Phone _____

Prices apply at Radio Shack Computer Centers and at participating stores and dealers. MS-DOS and monitor sold separately.
IBM/Registered TM International Business Machines Corp. MS and XENIX/TM Microsoft Corp. ViaNet/TM VIANETIX.

This Turbo PASCAL program turns your PC into an arcade game

Taking Turbo Powered Graphics for a Spin



When I received my Version 3.0 upgrade of *Turbo PASCAL*, I was very eager to try out some of the graphics capabilities which were not implemented under the Version 2.0 I had used for so long. In particular, I wanted to check out *Turbo*'s potential for animation by writing a small animated program. Since this was merely to be a trial, I thought I would keep the test program simple enough so it could be completed in a few hours, yet have fast-paced action during play. My experiment would let me know whether *Turbo* would give me a way to write longer, more sophisticated arcade games without resorting to the tedium of assembly language.

The project I decided on was the production of my own version of a famous arcade game, which involves having a ball burst through multicolored walls. However, there are some unique features to my version. I call the game *Smash About*.

Some differences from the traditional game are in the motion of the ball when it hits the wall under certain unusual circumstances, but I'll let the reader discover these characteristics on his or her own.

There is another personal twist I added. When the ball reaches a certain distance between the paddle and the wall, a small, random horizontal displacement is added to its motion.

Before the actual game begins, the player is allowed to choose one of four levels of difficulty — the higher the level, the faster the ball moves. After the player has decided which level to play, the playing field is drawn. The colored walls are at the top of the screen, while the paddle is at the bottom.

Soon after the completion of the display, a ball appears at a random height on the left or right of the screen. When the ball hits the wall, it knocks a "block" out of it. The player should move the paddle under the ball before it reaches the bottom of the screen to insure that the ball can bounce back up to the wall; otherwise, the ball will be lost. The game is over as soon as 10 balls have been lost.

It is necessary for the person playing the game to anticipate where the ball is *going to be* if the player is to be successful at keeping the ball in play. Two characteristics of the game account for this necessity. First, whenever you hit the 'Z' or '/' key to move the paddle, there is a fraction of a second delay before there is any movement. For that reason the player should hold down a key whenever possible, since that makes it repeat with no delay after the initial delay. Second, I purposely made the paddle move a little slower than the ball (you didn't want it to be easy, did you?). Even given those two limitations I have found that a player can keep each ball in play a fairly long time by moving the paddle in the direction the ball appears to be headed.

Readers who are interested in learning PASCAL may wish to study the procedures in the program to see how they work. More advanced programmers may want to add their own features or incorporate some of the techniques I have used in their own programs. For those who are just interested in playing the game — good luck! □

(Rick Boozer is the lead programmer at Frontier Electronics in Greenville, South Carolina. He has taught computer training classes for Radio Shack and programming at the University of South Carolina and holds a bachelor's degree in geology.)

The listing:

```
program SmashAbout;

{$I GRAPH.P}

type
    PadBuf=array[1..7Ø] of byte;BallBuf=array[1..8] of byte;BlankBuf=array[1..8Ø
] of byte;

var
    Answer:char;
    PaddlePos,BallPosX,BallPosY,StepX,StepY,Speed:integer;
    Paddle:PadBuf;
    Ball,EraseBall:BallBuf;
    Blank:BlankBuf;

procedure DrawLayers;

var
   Layer,x,y,y2,Color:integer;

begin
     Color:=Ø;
     for Layer:=1 to 6 do
        begin
             y:=4Ø+(Layer-1)*1Ø;y2:=y+9;Color:=Color+1;if Color>3 then Color:=1;
             for x:=Ø to 319 do
                begin
                     Draw(x,y,x,y2,Color);
                end;
        end;
end;


procedure DrawPaddle;

begin
     Draw(145,199,176,199,3);GetPic(Paddle,134,199,185,199);
end;


procedure DrawBall;

begin
     Plot(Ø,199,3);Plot(1,199,3);GetPic(Ball,Ø,199,1,199);GetPic(EraseBall,Ø,Ø,1
,1);
     PutPic(EraseBall,Ø,199);
end;


procedure CreateBlank;

begin
     GetPic(Blank,Ø,1Ø,1Ø,Ø);
end;


procedure ChooseBallStart(Var BallPosX,BallPosY,StepX,StepY:integer);
```

```
var Side:real;

begin
     BallPosY:=100+Random(90);StepY:=-1;Side:=Random;
     if Side<0.5 then begin
       BallPosX:=6;StepX:=1;
     end else begin
       BallPosX:=314;StepX:=-1;
     end;
end;


procedure CheckForVLayer(Var BallPosX,BallPosY,Score:integer);

var
   BlockPosX,BlockPosY:integer;

begin
     if (BallPosY+1>=35) and (BallPosY-2<=105) then
        begin
             BlockPosX:=(BallPosX DIV 10)*10;
             if StepY<=-1 then BlockPosY:=((BallPosY+16) DIV 10)*10
                else BlockPosY:=((BallPosY) DIV 10)*10;
             PutPic(Blank,BlockPosX,BlockPosY);
             PutPic(BlanK,BlockPosX,BlockPosY);
             Score:=Score+10;GotoXY(30,1);Write(Score);
        end;
```

```
end;

procedure CheckForHLayer(Var BallPosX,BallPosY,Score:integer);

var BlockPosX,BlockPosY:integer;

begin
     BlockPosY:=((BallPosY) DIV 1Ø)*1Ø+1Ø;
     if (BallPosY+1>=35) and (BallPosY-2<=1Ø5) then
        begin
             if GetDotColor(BallPosX-1,BallPosY)<>Ø then
                begin
                     BlockPosX:=((BallPosX) DIV 1Ø)*1Ø-1Ø;
                     If BlockPosX>=1Ø then
                     begin
                          PutPic(Blank,BlockPosX,BlockPosY);
                          Score:=Score+1Ø;GotoXY(3Ø,1);Write(Score);
                     end;
                end else begin
                     BlockPosX:=((BallPosX+2) DIV 1Ø)*1Ø+1;
                     If BlockPosX<=31Ø then
                        begin
                             PutPic(Blank,BlockPosX,BlockPosY);
                             Score:=Score+1Ø;GotoXY(3Ø,1);Write(Score);
                        end;
                end;
        end;
end;

procedure Beep;

begin
     Sound(1ØØØ);Delay(1ØØ);NoSound;
end;

procedure CheckForHit(var StepX,StepY,Score:Integer);

begin
     if (GetDotColor(BallPosX,BallPosY+1)<>Ø) or (GetDotColor(BallPosX+1,BallPos
Y+1)<>Ø)
        or (GetDotColor(BallPosX,BallPosY-2)<>Ø) or (GetDotColor(BallPosX+1,Ball
PosY-2)<>Ø) then
        begin
             StepY:=-StepY;CheckForVLayer(BallPosX,BallPosY,Score);
             if BallPosY>19Ø then
                if (GetDotColor(BallPosX+12,BallPosY+1)<>Ø) and (GetDotColor(Bal
lPosX-12,BallPosY+1)<>Ø) then
                   StepY:=StepY DIV Abs(StepY)*2
                else StepY:=StepY DIV Abs(StepY);
             Beep;
        end;
     if (GetDotColor(BallPosX-1,BallPosY)<>Ø) or (GetDotColor(BallPosX+2,BallPos
Y)<>Ø) then
        begin
             if BallPosY>1Ø5 then StepX:=-StepX;
             CheckForHLayer(BallPosX,BallPosY,Score);
             Beep;
        end;
```

```
     if (GetDotColor(BallPosX-1,BallPosY-1)<>Ø) and (BallPosY<19Ø)
        or (GetDotColor(BallPosX+2,BallPosY-1)<>Ø) and (BallPosY<19Ø) then
           PutPic(Blank,BallPosX DIV 1Ø*1Ø,BallPosY DIV 1Ø*1Ø);
     if (GetDotColor(BallPosX-1,BallPosY+1)<>Ø) and (BallPosY<19Ø)
        or (GetDotColor(BallPosX+2,BallPosY+1)<>Ø) and (BallPosY<19Ø) then
           PutPic(Blank,BallPosX DIV 1Ø *1Ø,BallPosY DIV 1Ø*1Ø+1Ø);
end;


procedure MoveBall(var Speed,Score:integer);

var OldBallPosX,OldBallPosY:integer;

begin
      OldBallPosX:=BallPosX;OldBallPosY:=BallPosY;
      CheckForHit(StepX,StepY,Score);
      if (BallPosX<3) or (BallPosX>315) then
         begin
              StepX:=-StepX;Beep;
         end;
      if BallPosY<13 then
         begin
              StepY:=-StepY;Beep;
         end;
      BallPosX:=BallPosX+StepX;BallPosY:=BallPosY+StepY;
      Delay(Speed);
      if (BallPosX>1Ø) and (BallPosX<3Ø4) and (BallPosY=1Ø5) then BallPosX:=Ball
PosX+Random(9)-5;
      PutPic(Ball,BallPosX,BallPosY);
      PutPic(EraseBall,OldBallPosX,OldBallPosY);
end;


procedure Pause;

begin
     delay(1ØØØ);
     Repeat
     Until (KeyPressed=true);
end;


procedure MovePaddle;

var
   ch:char;chbyte:byte;

begin
     if KeyPressed then begin
        Read(kbd,ch);chbyte:=ord(ch);
        if (chbyte<>122) and (chbyte<>9Ø) and (chbyte<>47) and (chbyte<>44) then
 Exit;
        if (chbyte=44) then Pause;
        if (chbyte=122) or (chbyte=9Ø) then PaddlePos:=PaddlePos-9;
        if (chbyte=47) then PaddlePos:=PaddlePos+9;
        if PaddlePos<-9 then PaddlePos:=-9;
        if PaddlePos>279 then PaddlePos:=279;
     end;
     PutPic(Paddle,PaddlePos,199);
```

```
end;

procedure CheckForMiss(Var Flag:integer);

begin
     if BallPosY>=199 then Flag:=1 else Flag:=Ø;
end;


procedure CheckForClear(var Speed:integer);

var x,x2,y,PointFlag,Colors:integer;

begin
     PointFlag:=Ø;
     for y:=4Ø to 1ØØ do
         begin
              for x:=Ø to 31 do
                  begin
                       x2:=x * 1Ø+2;
                       if GetDotColor(x2,y)<>Ø then PointFlag:=1;
                  end;
         end;
     if PointFlag=Ø then
        begin
             Colors:=Colors+1;if (Colors>=4) or (Colors<Ø) then Colors:=Ø;
             Palette(Colors);DrawLayers;Speed:=Speed-1;
        end;
end;


procedure PlayGame;

var
   Flag,Times,Score:integer;

begin
     Randomize;Score:=Ø;Palette(2);
     PaddlePos:=134;Times:=Ø;
     Repeat
           ChooseBallStart(BallPosX,BallPosY,StepX,StepY);
           MoveBall(Score,Speed);
           Repeat
                 MoveBall(Speed,Score);
                 MovePaddle;
                 CheckForMiss(Flag);if Flag=1 then Times:=Times+1;
           Until Flag=1;
           PutPic(EraseBall,BallPosX,BallPosY);
           Sound(1ØØ);Delay(1ØØ);NoSound;CheckForClear(Speed);
     Until Times=9;
end;


procedure TitleScreen;

var
   ch:char;

begin
```

WE'RE BRINGING THE TANDY

PCM's BROADENING ITS SPECTRUM

PCM and the Delphi Information Utility have joined together to allow Tandy MS-DOS owners all over the world to connect with one another!

Delphi is a full-service information utility. It offers everything from up-to-the-minute news stories from The Associated Press to electronic mail services. But, *best of all*, it now has **a special forum for owners of MS-DOS computers, and it's operated by the people who bring you PCM each month.**

The MS-DOS Special Interest Group (SIG) features a variety of services, including an open forum where you can send and receive messages from Tandy owners all over the world. It also has databases for exchanging programs, where you can upload your favorites and download programs written by other Tandy MS-DOS enthusiasts.

When setting up your account with Delphi, if you do not have a credit card or prefer not to use it, Delphi requires that you send $20 to give your account a positive balance. This will be refunded after your first free hour if you choose to no longer use the system or it will be applied to future connect charges.

THE WORLD OF MS-DOS

The MS-DOS SIG's conference feature allows you to meet electronically with other members of the Tandy Community. You can join conferences with many notables in the world of MS-DOS on a regular basis. Conference schedules will appear in PCM each month. Be sure to check online announcements for changes and additions.

PCM ONLINE

On Delphi, You can renew your PCM subscription, make a fast and easy order for software or hardware from a multitude of vendors, or inquire about products on the MS-DOS SIG.

We also have a number of **programs that you can download** and use, just for the cost of the time you spend transferring them. There'll also be **corrections for PCM articles**, helpful hints and many other useful features.

FREE LIFETIME MEMBERSHIP

PCM is offering subscribers a free lifetime subscription to Delphi — a $29.95 value — and a free hour of connect time — a $6 value at either 300 or 1200 Baud during evening, holiday and weekend hours — so you can sample Delphi and the brand new PCM MS-DOS SIG. **That's right. Your subscription to PCM entitles you to this $35.95 value as a *free bonus!***

If you're not a PCM subscriber, just enter your order when you sign on with Delphi and you'll get the same great deal! For our $28 subscription fee, you'll get the finest Tandy computer magazine ever, a free lifetime subscription to Delphi and a free hour of connect time.

SAVE EVEN MORE

Want to save an extra $15? While you're online, you can order the *Delphi Handbook and Command Card* ($21.95) and three hours of connect time ($18) for only $24.95.

Delphi provides us all with **Immediate Tandy Community**. Check it out today. After all, you can sample it for free!

Problems? Call Delphi at 1-800-544-4005.

DELPHI

TYPE: GROUP MSDOS

COMMUNITY TOGETHER!

How to reach the MS-DOS SIG on Delphi . . .

There are several ways to connect to Delphi and PCM's MS-DOS SIG. In most cities you will not even have to pay long distance charges; you can use special data communications networks, like Uninet, Tymnet and the Canadian Datapac network.

First, set your terminal program to operate at either 300 or 1200 Baud (depending on the modem you have), and also select either 7 bits with even parity or 8 bits with no parity, and 1 stop bit. (If one combination doesn't work, try another.)

Decide which network you should use. There is no surcharge for Uninet or Tymnet. Canadian residents using Datapac will be charged an additional $3 (U.S.) per hour.

On Uninet: Call (800) 821-5340 to get the Uninet number for your area. After you call the appropriate number for your own area and make connection, press ENTER, the period key (.) and ENTER again. At the "service:" prompt, type GVC (for General Videotex Corporation) and ENTER.

On Tymnet: Call (800) 336-0149 to get the Tymnet number for your area. After you dial your designated number and make connection, simply press 'A' no matter what appears on the screen. When "please log in:" appears, type DELPHI and press ENTER.

From Canada (on Datapac): Call Delphi Customer Service at (617) 491-3393 to get the Datapac number for your area. After you connect, press the period key (.) and ENTER (use two periods if you're using 1200 Baud). Type SET 2:1, 3:126 and press ENTER. Now type p 1 3106, DELPHI; and press ENTER. Delphi will bill you an additional $3 (U.S.) per hour for Datapac connection surcharges.

From other countries: Many countries have their own data networks that can connect to either Uninet or Tymnet. Check with the telephone authorities in your country for details on how to sign up for this service. When you have an account set up, you can reach Delphi with a "host code" of 312561703088 through Uninet, or 310600601500 through Tymnet. (You'll have to pay the toll charges for this connection.)

Type in Your User Name

If you're already a subscriber to PCM, at the "USERNAME:" prompt, type PCMSUB and press ENTER. At the "PASSWORD:" prompt, type your individual subscription number from the mailing label of your latest issue of PCM. (If there are one or more zeros at the beginning of this number, include them.)

If you don't already have a subscription, at the "USERNAME:" prompt, type PCMORDER and press ENTER. At the "PASSWORD:" prompt, type SENDSUB and press ENTER. Have your MasterCard, VISA or American Express card ready, because you'll be led through a series of questions that will enable us to put your PCM and Delphi subscriptions into effect. In an effort to hold down non-editorial costs, we do not bill for subscriptions.

If you make a typing error, just press ENTER and start over. Remember that at any point, when you're on Delphi, you can type HELP to get help on how to use the system. To get off the system just type BYE.

If you find that you're unable to log onto Delphi and enter the MS-DOS SIG after following these instructions, call us during afternoon business hours at (502) 228-4492. We'll be glad to offer assistance.

Come Visit Us! Type: GROUP MSDOS

After you sign in, you'll be prompted to set up your own, personal "user name" — Delphi is a friendly service, no numbers to remember — and you'll be asked a number of questions so Delphi can set up your account. You'll also be assigned a temporary password. No time is assessed against your free hour of service while you answer these questions.

Delphi will tell you that your account will be ready after 6 p.m. the same day if you sign up before noon (Eastern time zone.) If not, your account will be ready at 6 p.m. the next day. Once an account is opened, *each PCM subscriber will be credited with an hour of free time!*

When you log back in, use your chosen user name and your temporary password to access the system. At that point, you will meet Max, who will help you configure things and will change your temporary password into your own *personal* password. This is the password you will use for subsequent sessions — or until you change it. Again, there is *no charge* for the time you spend configuring your account. Your hour of free time starts after Max bids you goodbye.

You'll wind up at the Delphi Main Menu; type in GROUP MSDOS and **join us on the MS-DOS SIG!**

$99 for a year's subscription to PCM ON DISK!

Each and every month, PCM offers an assortment of programs for the home and office — utilities, business applications, games and more — all for the price of typing in the listings.

If you're a person on the move, and too busy to spend time typing lengthy listings, then PCM has just what you need — **PCM ON DISK.**

PCM ON DISK provides each issue's MS-DOS programs ready to load and run with only a few keystrokes!

AS AN ADDED BONUS, **PCM ON DISK** will include PCM's portable computer programs, which may be downloaded to portables for those readers who have both an MS-DOS machine and one of Tandy's portable computers.

That means you get a disk each month for a year, containing all programs listed in that month's issue, for an average cost of only $8.25!* We think it's worth your time to subscribe.

*Disks for individual issues (beginning this month) may be ordered for only $12 each.

PCM

PCM ON DISK
The Falsoft Building
P.O. Box 385
Prospect, KY 40059
(502) 228-4492

I would like a one-year (12 months) subscription to PCM ON DISK.

Name _____

Address _____

City _____ State _____ ZIP _____

Payment Enclosed ☐

Charge Visa ☐ MasterCard ☐ American Express ☐

My Account# _____ Card Exp. Date _____

Signature _____

*Subscriptions to PCM ON DISK are $99 a year in the United States. Canadian rate U.S. $115. All other countries U.S. $115; PCM ON DISK single copy rate is: $12 within the U.S.; U.S. $14, Canadian rate; U.S. $16, all other countries. Kentucky residents add 5% sales tax. All subscriptions begin with the current issue unless otherwise specified. Please allow 5 to 6 weeks for first copy. **U.S. currency *only*, please.** In order to hold down non-editorial costs, we do not bill.

```
      ClrScr;TextMode(C8Ø);
      TextColor(4);GotoXY(34,1Ø);Writeln('* SMASH ABOUT *');
      TextColor(2);GotoXY(12,12);
      Writeln('a new version of a familiar arcade game for PC Compatibles');
      TextColor(1);GotoXY(35,15);Writeln('by Rick Boozer');
      GotoXY(32,16);Writeln('Phone: 8Ø3-268-77Ø7');
      TextColor(8);GotoXY(9,24);
      Writeln('Hold down the Z and / keys to move the paddle, comma pauses game')
;
      Readln(ch);
end;


procedure SelectLevel(var Speed:integer);

var
   Answer:char;

begin
      repeat
            ClrScr;
            Writeln;Writeln;
            Writeln('Ø: Easy');
            Writeln('1: Novice');
            Writeln('2: Intermediate');
            Writeln('3: Expert');
            GotoXY(1,12);Writeln('Choose difficulty level: Ø, 1, 2, 3');
            Readln(Answer);
      Until Answer in ['Ø','1','2','3'];
      case Answer of
            'Ø':Speed:=11;
            '1':Speed:=6;
            '2':Speed:=4;
            '3':Speed:=3;
      end;
ClrScr;
end;


begin
      Repeat
            TitleScreen;
            SelectLevel(Speed);
            ClrScr;TextMode(BW4Ø);GraphColorMode;Palette(2);
            DrawLayers;
            DrawPaddle;
            DrawBall;
            CreateBlank;
            PlayGame;
            Repeat
                  GotoXY(1,1);
                  Writeln;
                  Write('Another game?');
                  ClrEOL;
                  Readln(Answer);
            Until UpCase(Answer) in ['Y','N'];
      Until UpCase(Answer) in ['N'];
      ClrScr;
end.
```

PCM

It's the Holmes Engineering/PCSG "chipmunk"

PORTABLE Disk Drive

AS FAST AS RAM

for the Model 100/200

NEW LOW PRICE
$499.

Includes a library of six powerful programs

NOW AVAILABLE ON THE M-200

ASK ABOUT BUDGET MODEL $399.

Uses the main menu concept. You see the disk directory instantly, arranged on your M-100 screen like your main menu. Just move the widebar cursor and transfer files with a function key. You can run a file directly from the diskette with the ENTER key. Uses 3½" microfloppy diskettes that have a rigid plastic casing and a metal core. They're tough and nonflexible. You can carry several in a shirt pocket without damage. There's 358K on a diskette. Ten of these in your briefcase and you've got 3½ megabytes.

Drive weighs only three lbs. and it works directly from the 110 outlet and recharges at the same time. It recharges in six hours with thousands of pages transferred between charges. It's compact, with dimensions of 2¼" x 5½" x 7.5"; and fits easily into your briefcase along with your Model 100 or 200.

Machine code programs, BASIC programs, *Lucid* files and documents all are saved and retrieved with no protocol–instantly, ready to run.

In a special association, Holmes Engineering and PCSG have worked together combining the hardware knowledge of Holmes and the software expertise of PCSG. The result is a product that can only be regarded as excellent.

You see the disk directory instantly; works just like the main menu

Here is what is really exciting. The portable disk drive has Random Access. Included as part of the operating system in the drive (ROM) is a very powerful disk BASIC.

This means that you can have BASIC programs that will access the diskette and read and write records directly on the diskette.

Just imagine yourself with this kind of capability.

Database–The portable disk drive stores your mailing list, inventory items, part #s and descriptions or any other data that you need to recall.

358K on a diskette

Invoice (purchase order)–At the touch of a button you can print out your sequentially numbered, professionally done invoices. This is truly professional invoicing capability.

Purchase orders are just as easy.

Sort – This excellent utility allows you rapid sorting of any records you have compiled. You can write the newly sorted list back in the same file on the diskette or to a new file.

Telecom interface–If you are a user that likes to access other computers or databases (for example CompuServe) by telephone then this powerful facility alone is worth the price of the disk drive. You can automatically download and upload information onto the diskette.

Calendar – Everyone who has seen this program has said, "This is the first calendar/diary/scheduler on any computer anywhere that I can use. It is so functional."

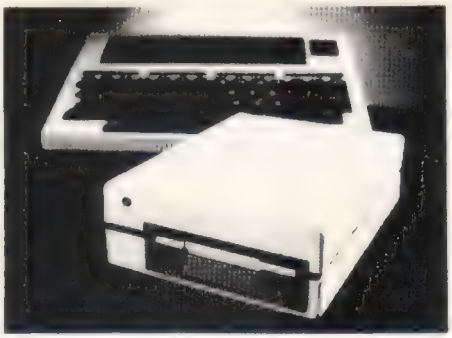
The calendar program is usable for two reasons, first it is designed correctly, and second you have the memory (358K) on the diskette to log and access a tremendous amount of notes over a long period of time.

Personal Finance Manager – This wonderful program truly lets you keep track of your finances.

All your records are kept on the diskette. Bank accounts (checking and savings) and charge accounts such as MasterCard and Visa.

We at PCSG believe we have the ultimate Model 100 system, the Portable Disk Drive plus the *Lucid* spreadsheet on snap-in ROM, *Write ROM* word processing and the new 64K RAM expansion now available from PCSG.

We want you to find out for yourself at no risk. If you aren't totally satisfied within 30 days, simply return the disk drive for a full refund. Priced at $499.00, including the software library. MasterCard, Visa, COD.



PORTABLE COMPUTER SUPPORT GROUP

1-214-351-0564

11035 Harry Hines Blvd., #207
Dallas, Texas 75229

PCSG offers a 30-day/money-back trial!

Using the Spreadsheet As A Scheduling Tool

By Richard A. White
PCM Contributing Editor

A key project planning and management tool is the schedule. There are a fair number of project management and schedule-making packages available for MS-DOS machines. These are generally fairly expensive, though, and may not meet the needs for smaller projects or those in the early planning stages.

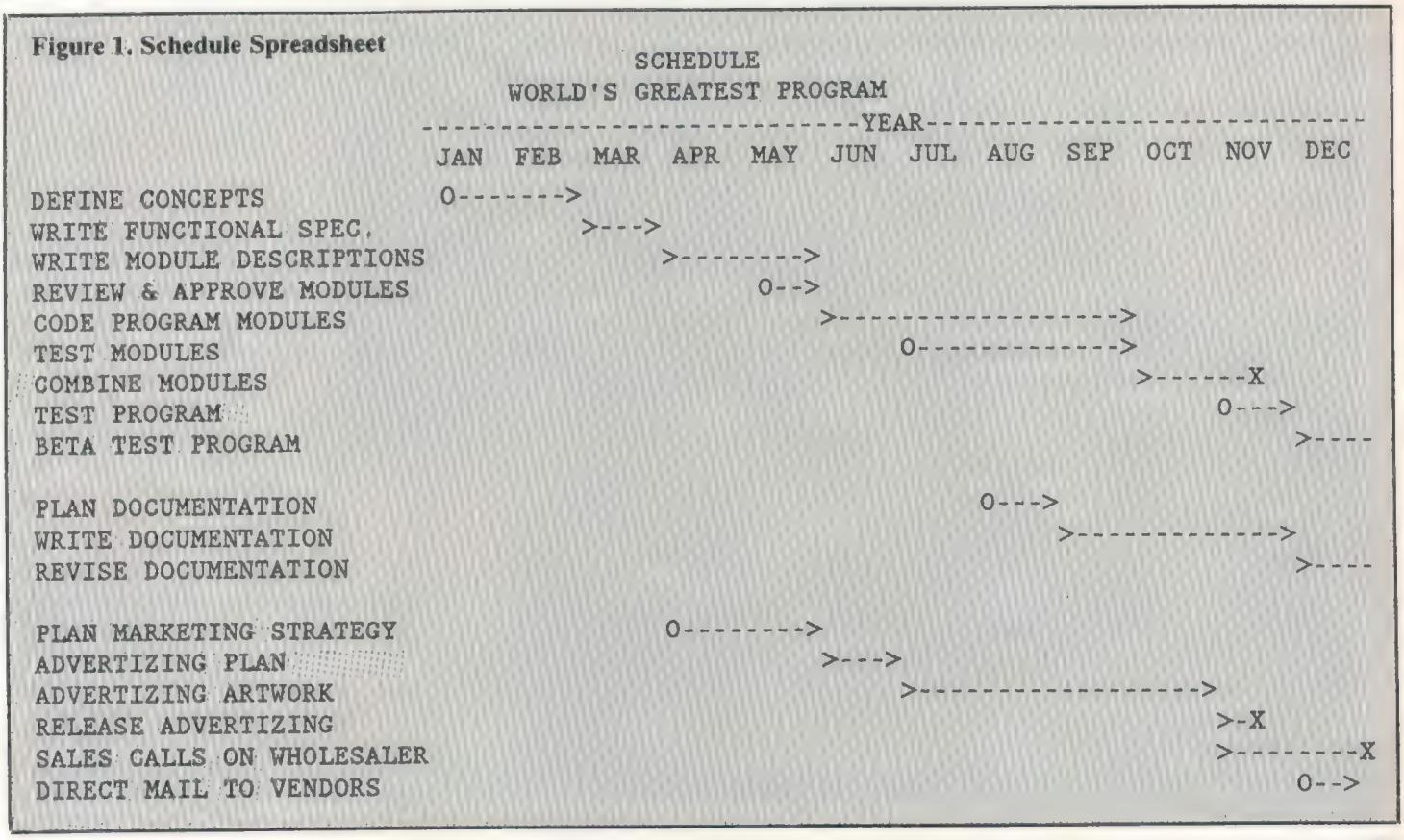
(Richard White has a long background with microcomputers and specializes in BASIC programming. He has authored numerous programs and articles. His work has appeared in PCM's sister publication, THE RAINBOW.)

One of the earliest such program was *VisiSchedule*, which I have used a time or two. However, I recently had a need that *VisiSchedule* did not satisfy. Basically, I wanted to make a schedule by month extending out over 21 months and put it on a single 11-by-15 sheet. This was a development project with unknowns, and a day-by-day or even week-by-week schedule was ridiculous. Yet these were the only two options available in *VisiSchedule*.

Obviously, capabilities like this might be available in some other programming package. As it turned out, I could do just what I wanted in a spreadsheet. The spreadsheet serves as a structure for the schedule and as a data entry and editing tool. Calculation is involved only in the second step, the effort estimate, and even there it is minimal. The application is so simple that I need do little more than show a couple of examples with a few comments and let you find your own scheduling applications.

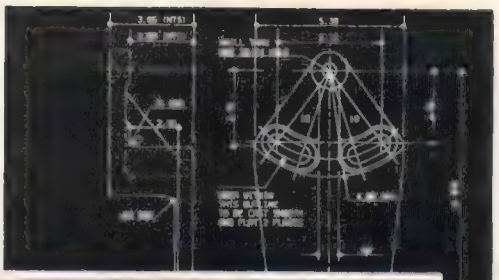
The example is a fabrication, though it does show steps that might be reasonably involved in generating a piece of commercial software, including getting the advertising out well before the package could be ready for sale under the best circumstances. I like my examples to be representative of real life.

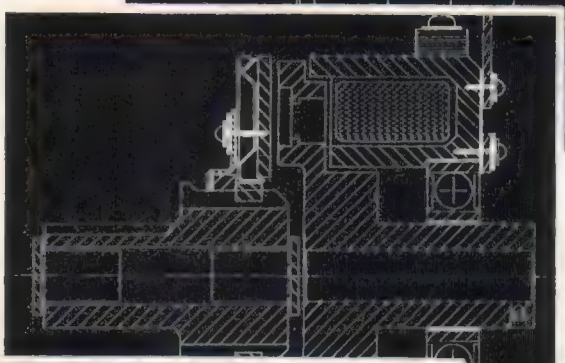
Figure 1. Schedule Spreadsheet

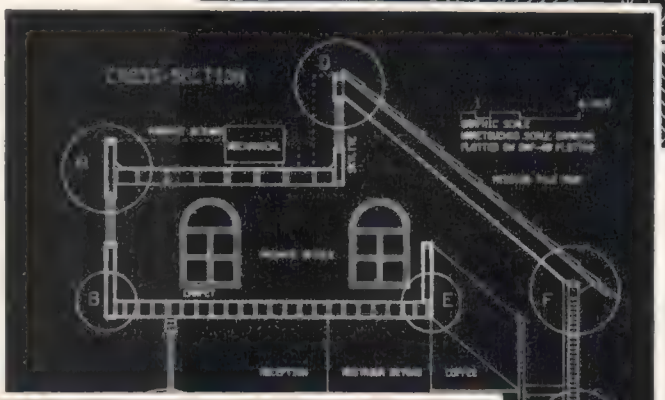


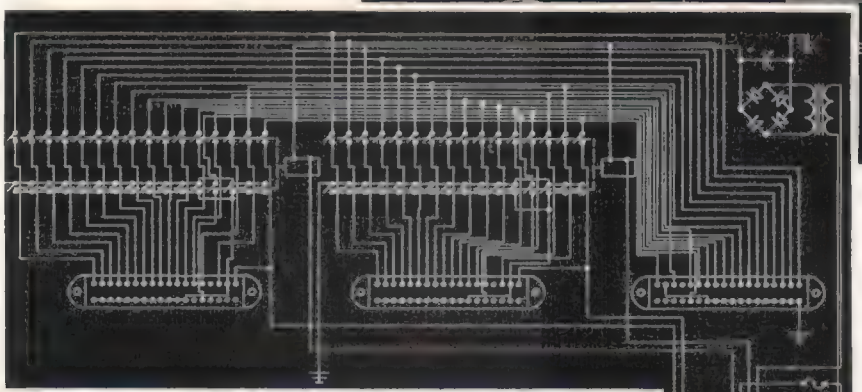
Models III 4 4D 4P 1000 1200
Computer Aided Drafting

Excellent system benefit / cost ratio
Drawing sizes up to 24" x 36"
Two full size overlays
Direct and auto zoom and pan
Decimal and "customary" scaling systems
Full inch and fractional grids
Cursor snap to grid or intra-grid
Variable cursor speed control
Single key functions from keyboard
Mnemonic function symbols easy to learn
Single operating mode at all times
On-line help screen and memory alert
Accuracy up to 1/160 or an inch
Digital (coordinate) input functions
Block copy and erase any time
Block reduce / enlarge 1% through 9999%
Block rotate 1 through 360 degrees
Merge, rotate, rescale drawings
Save / load drawings to / from disk
Cut out and save parts of drawings
Text labels full upper / lower ASCII
Two text sizes, two directions
Text duplication saves memory / typing
Left justified "typewriter" text option
Hardcopy reduce / enlarge 1% - 9999%
All hardcopy media (paper, mylar, ink)









T.CAD

Computer Aided Drafting systems for architects, engineers, educators, manufacturers, builders, students ...

MICRODEX
1212 North Sawtelle
Tucson, Arizona 85716
Phone (602) 326-3502

Tandy may be the only company in the business that does not blow the horn more than a month or two before the goods are out.

First, set the global column width to '5'. Each month column now becomes five characters wide. Next set column A individually to 25 or 30. This overrides the global column setting. Now, type in the sheet headings and enter the month names at the top of the month columns. I used three character contractions set in one space, which centers the month name on each column. The year can be put in as shown if you want. You will want to do this on multi-year schedules. This format fits 21 months across on a 15-inch page, more with condensed type. Our example project might eventually need that if the programmers get into hot water.

The effort areas are typed down column A. This column and the heading including the month names should be put into windows so they stay on the screen as you move around the spreadsheet. In *Lotus 1-2-3* you have a choice of either "Windows" or "Title" from the Worksheet menu. For what we want, Titles is easier. On entry of the command /WT, the following menu appears:

```
Horizontal Vertical Both Clear
```

If you select Horizontal, the rows on the screen above the cell pointer are locked in place. If you select vertical, the column or columns to the left of the cell pointer are frozen. "Both" is self-explanatory and is what we want. Move the cell pointer to column B in the row below the cell names and press B. Your horizontal and vertical titles will stay on the screen and scroll in sync with the data as you move about the spreadsheet.

Now, you will certainly want to get back into column A to add more effort descriptions. In *Lotus*, use F5, the GOTO function key and enter A and the row number where you wish to add text or edit.

The schedule lines are essentially labels. There are two originators. An 'O' means that a task is arbitrarily begun without depending on the completion of an immediately previous task. This type of task is not critical path, though it may depend on some task being completed some time before. A greater-than symbol (>) beginning a task line means that the task cannot begin until one or more immediately prior tasks are complete. The critical path through the schedules consists of the chain of these types of tasks.

Likewise, there are two task termination symbols. Non-critical path tasks are terminated with an 'X'. Critical path tasks end with the greater-than symbol again pointing to there being a next dependant task.

Since month columns are five characters wide, you can easily discriminate early-, mid- and late-month endings. For this type of application, finer distinctions are not needed.

Save and print your schedule. Now you can go back into your copy in memory and type in effort estimates for each task over the months the effort occurs. The schedule lines are replaced with effort days. Use @SUM(B7 . . . B26)/20 to sum a column and express total monthly effort as effort months. This will help down the road when you have to explain to the boss that you and your assistant got behind because the two of you couldn't get the four effort months of work predicted for last month completed. Save your effort prediction under a name different from the one you used to save the schedule. An example is shown in Figure 2. □

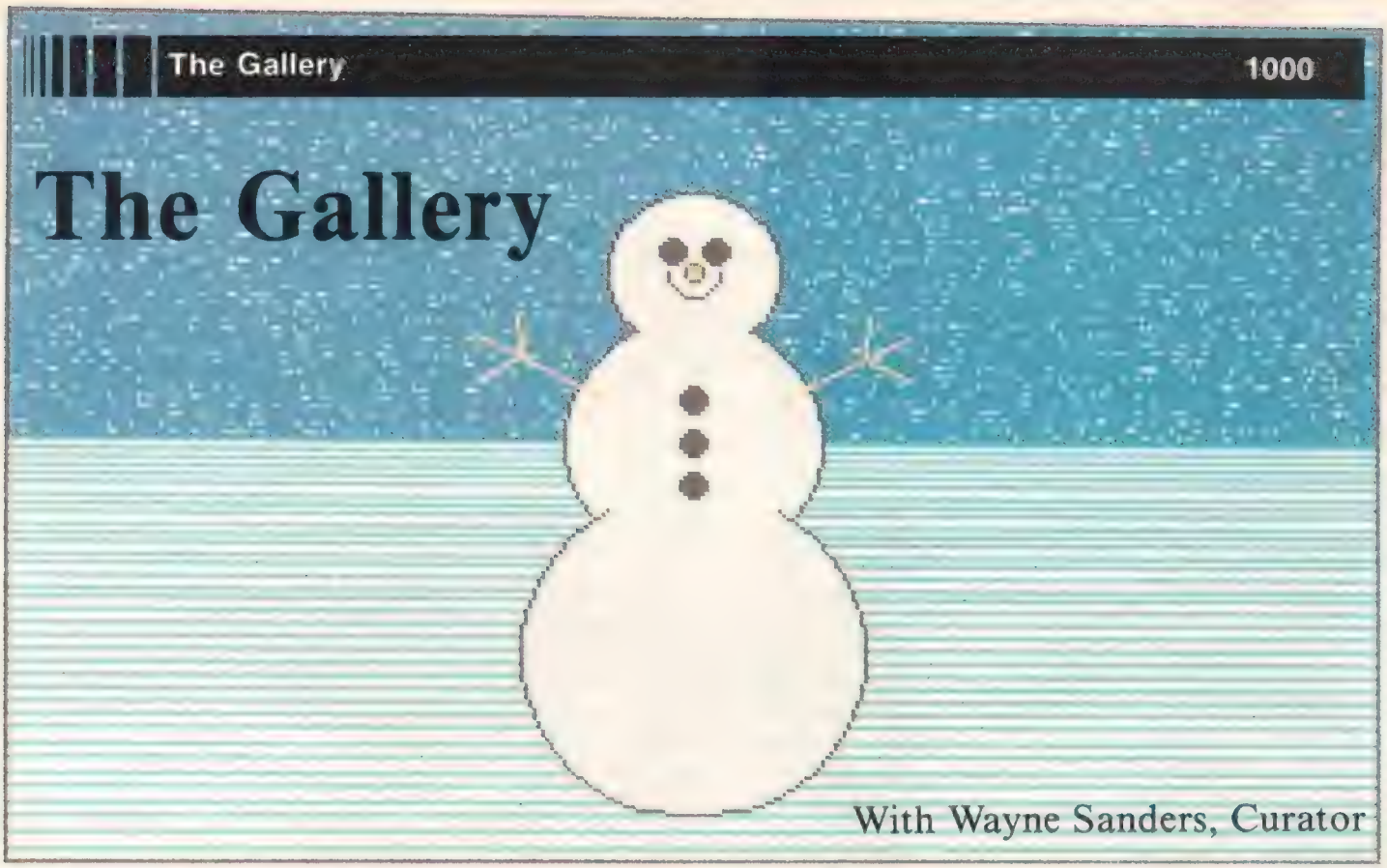
Figure 2. An Example of an Effort Prediction on the Schedule Shell.

EFFORT ESTIMATE
WORLD'S GREATEST PROGRAM

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
DEFINE CONCEPTS	15	15										
WRITE FUNCTIONAL SPEC.			2Ø									
WRITE MODULE DESCRIPTIONS				4Ø	4Ø							
REVIEW & APPROVE MODULES					1Ø							
CODE PROGRAM MODULES						6Ø	6Ø	6Ø	6Ø			
TEST MODULES							1Ø	2Ø	2Ø			
COMBINE MODULES										2Ø	1Ø	
TEST PROGRAM											2Ø	
BETA TEST PROGRAM												5
PLAN DOCUMENTATION								15				
WRITE DOCUMENTATION									25	25	25	
REVISE DOCUMENTATION												2Ø
PLAN MARKETING STRATEGY				15	15							
ADVERTIZING PLAN						25						
ADVERTIZING ARTWORK							2Ø	2Ø	2Ø	2Ø		
RELEASE ADVERTIZING											15	
SALES CALLS ON WHOLESALERS											2Ø	2Ø
DIRECT MAIL TO VENDORS												15
EFFORT MONTHS PER MONTH	.75	.75	1	2.75	3.25	4.25	4.5	5.75	6.25	3.25	4.5	3

PCM

The Gallery



With Wayne Sanders, Curator

This month's featured Gallery exhibit is a typical winter scene — a snowman standing happily in a raging blizzard. When the program is run on your Tandy 1000, the picture comes to life as the snow seems to fall from the sky.

Our thanks go out to Jim Fahl of Springfield, Illinois for this wonderful wintery treat.

If you would like to have your computer creation presented here, send it in. A winning gallery exhibit is chosen each month and the "artist" is awarded $50. Address your entries to PCM Gallery, P.O. Box 385, Prospect, KY 40059.

The listing:

```
1000 CLEAR ,,,32768!:SCREEN 5:CLS:KEY ON:KEY OFF:TS=1.745329E-02
1010 PALETTE 2,7:PALETTE 5,7:PALETTE 9,7
1020 LINE (0,0)-(319,99),1,BF:LINE (0,100)-(319,199),7,BF
1030 FOR Y=0 TO 100 STEP 6
1040    FOR J=1 TO 50
1050       PSET (RND*360,Y),2:PSET (RND*360,Y+2),5:PSET (RND*360,Y+4),9
1060    NEXT J
1070 NEXT Y
1080 Y=100:WHILE Y<200:LINE (0,Y)-(319,Y),3:Y=Y*1.02:WEND
1090 CIRCLE (160,150),40,8,TS*120,TS*60:CIRCLE (160,100),30,8,TS*320,TS*65
1100 CIRCLE (160,100),30,8,TS*115,TS*220:CIRCLE (160,60),20,8,TS*310,TS*230
1110 PAINT (160,100),15,8
1120 CIRCLE (155,55),3,0:CIRCLE (165,55),3,0
1130 PAINT (155,55),0,0:PAINT (165,55),0,0
1140 CIRCLE (160,90),3,0:CIRCLE (160,100),3,0:CIRCLE (160,110),3,0
1150 PAINT (160,90),0,0:PAINT (160,100),0,0:PAINT (160,110),0,0
1160 CIRCLE (160,60),2,0:PAINT (160,60),6,0
1170 CIRCLE (160,60),6,0,180*TS,360*TS
1180 LINE (180,90)-(200,80),6:LINE (200,80)-(201,70),6
1190 LINE (200,80)-(210,85),6:LINE (200,80)-(210,75),6
1200 LINE (140,90)-(120,80),6:LINE (120,80)-(119,70),6
1210 LINE (120,80)-(110,85),6:LINE (120,80)-(110,75),6
1220 PALETTE 2,7:PALETTE 9,1:GOSUB 1250
1230 PALETTE 5,7:PALETTE 2,1:GOSUB 1250
1240 PALETTE 9,7:PALETTE 5,1:GOSUB 1250:GOTO 1220
1250 FOR J=1 TO 100:NEXT J:RETURN
```

PCM

No need to depend on the capricious vagaries of magic to develop truly fast text display; the trick lies in an accessible logic revealed here by the author

Superfast Displays

By John B. Harrell, III

Did you try my *Menu* program from the November and December issues of PCM? Do you wonder how that program and many others display text so fast? It is not really magic — you simply write directly to the video memory. I promised to reveal the secrets of the video display in future articles, and I will begin with the simple feat of displaying text.

I am very impatient — nothing dismays me more than waiting for a program to display its information. In fact, the first iterations of *Menu* were so slow that I almost abandoned the program's development. Other major software packages (*Lotus 1-2-3*, etc.) use the same techniques I will describe here and you can use them also to create spectacular displays.

The simple PASCAL program in Listing 1 provides an elementary windowing program that allows rapid text display. It is set up for an IBM PC (or close compatible) but it also contains all of the code necessary to make it run on the Tandy 2000. I will describe the IBM PC architecture here and add additional comments to cover the specific operation of the Tandy 2000.

Windows and Frames

Program Listing 1 provides a simple set of routines that will draw a "window" on the video screen in the location and colors specified. Other routines display text and headers and perform other system routines.

I used *Turbo* PASCAL for my program because it is easy as BASIC to use, super-quick and *cheap*! It is also self-documenting and you can use it as a program design language to transport algorithms from one language to another.

Program Listing 1 is set up to compile for the Tandy 1000 or any other IBM PC compatible computer. In this definition of compatibility, I mean the video display segments are in the same locations as on the IBM PC and that the BIOS code provides the interrupt services used in the program. Not all MS-DOS machines provide these interrupt services; notably Wang and Zenith.

Before I get into the program, some explanation of the video structure is required. The IBM PC maintains a RAM area beginning at B800H (B000H for the monochrome display) which is organized so every two bytes describes a location on the screen — one byte for the character followed by one byte for the character's attribute.

Using some simple arithmetic, 25 lines of 80 characters and attributes implies that 4,000 bytes of memory are used by one screen display. With the color graphics option, there are 16K bytes of video RAM available, which may be organized in up to four video "pages" in the 80 by 25 mode. Each page is individually selected by using a service interrupt to reset the video controller's memory fetch pointer. This program (and *Menu*) uses only the first video page.

Any character from the extended ASCII character set may be stored into any of these locations. The attribute byte requires more detailed information. I'll try to explain it using the data from Table 1.

The attribute byte contains two "nybbles" — the left nybble describes the character background attributes and the right nybble describes the foreground attributes. Table 1 presents the data for both the monochrome and color monitors. I will limit my discussion to the color monitor as this is the dominant mode of operation for the Tandy 1000.

The background attribute (left nybble) may contain a color number for any of the eight colors listed in Table 1. The upper bit of the left nybble is used to denote blink mode for the foreground character color. For example, if you select the background color green (Color 2) and want the foreground character to blink, you would encode the left nybble as 'A' (2 + 8 = 10 or 'A' in Hex).

The right nybble controls the color of the foreground character and the upper bit controls the brightness of the color. This allows characters to take on the full range of 16 colors available. For example, the program constant "chilite" in Listing 1 is initialized to the value 4E Hex.

Using these rules, characters written with this attribute will be displayed non-blinking on a red background (left nybble = 0 + 4 = 4) and each character will be in bright yellow (right nybble = 8 + 6 = E). If I had wanted these characters to blink, then this attribute would have been coded as CE instead of 4E.

The Tandy 2000 works in a very similar manner when it operates in the monochrome video mode. Video RAM is located in the upper 5K of the last active user RAM board and must be located each time the program is loaded. Attributes are stored and used similar to the Tandy 1000 *but* the actual video attributes bear no resemblance to the IBM PC compatible codes.

If you use the system MS-DOS or BIOS level display functions, you may use IBM codes — they are converted automatically. However, with my demonstration, you're on your own. Table 2 presents a description of the attribute

---

(John B. Harrell, III has written for microcomputer magazines for three years. He holds a bachelor's degree in computer science and is a software technical expert for Navy electronic support measures systems.)

Table 1

IBM PC Color Attributes

Attribute Byte

DKGD	FGND

Color Selections for Attribute Bytes

0 - Black (Gray)
1 - Blue
2 - Green
3 - Cyan
4 - Red
5 - Magenta
6 - Brown (Yellow)
7 - White

Notes: 1. Adding eight to the BKGD (background) attribute causes the foreground character to blink.
2. Adding eight to the FGND (foreground) attribute causes the foreground character to be displayed in high intensity mode. In the case of the colors black and brown, the colors in parentheses are displayed.

byte and the respective bit functions. Because of the method of implementation, the Tandy 2000 does not use these attributes in the color mode for background/foreground color generation.

Program Description

While you are entering program Listing 1, note the commented areas pertaining to the particular hardware structure. *Turbo* PASCAL has an exceptionally nice feature that allows using one type of comment delimiter (the "(*" and "*)" symbols) to offset any other block of text, including comments delimited with the braces "{ }." I use this feature to surround the blocks of code for the machine type I am *not* going to use. In Listing 1, all of the Tandy 2000 code is surrounded by these delimiters.

The first two routines, `off_cursor` and `on_cursor`, use the BIOS code video interrupt to selectively disable and enable the video cursor. The record `regs` is a variable providing the interrupt procedure initial values for the CPU registers. It also informs the interrupt routine where to store the return values, if there are any — we'll use this feature later.

The next two procedures, `cursor` and `getcursor`, overcome what I feel is an inconsistency in *Turbo*. These routines replace the *Turbo* PASCAL `GotoXY`, `WhereX` and `WhereY` routines. These routines assume the cursor position is referenced to the upper left corner at (1,1) as opposed to the system BIOS functions, which expect it to be referenced to (0,0).

The procedure `mode` simply sets the video mode according to the input parameter. This allows me precise control over the video mode without having to anticipate how your particular *Turbo* PASCAL version is installed.

Now, we begin with the heart of the video drivers. The first procedure, `init_video`, initializes the proper global constants for your hardware, clears the screen and turns off the video cursor. For the compatibles, this requires a check of the system hardware to see what type of monitor is installed. The constant definitions at the beginning of the program presume a color monitor configuration.

If the program detects a monochrome display, it will reset the segment address correctly. Next, it will change the attributes to the proper monochrome display values. Finally, it sets the video screen mode to 80 by 25 monochrome.

The Tandy 2000 code assumes it will always operate in the monochrome mode even on a color monitor. All that is really required is to locate the proper segment address for the beginning of the video monochrome RAM. After programming the four palette registers with their respective colors, the program issues a BIOS interrupt to request memory size. This interrupt returns the total number of kilobytes of RAM in the `AX` register. The proper paragraph segment is calculated from this returned value and stored in the global variable `video_mem`.

With one exception, `disp_str` simply displays the string at the appropriate row and column locations. The characters are stored in the colors defined by the global attribute variable `attrib`. Memory absolute addresses are accessed using the *Turbo* PASCAL `Memw` statement and this is similar to using a `DEF SEG` and `POKE` statement in BASIC.

Note that the attribute is assembled as the high order byte and the actual character is placed in the low order byte of the 16-bit word to be stored. This is because of the Intel method of storing the high order byte in the higher-addressed position. Remember, we want a character byte followed by its attribute byte, and that is the way this character attribute pair will be stored.

The exception to simplicity in this routine arises from a hardware limitation of the IBM PC. Due to bus contentions between the CPU and the video controller, if you try to write characters to the video RAM segment indiscriminately, you will see flickers and bursts of "hash" all throughout your screen until the writing stops.

The `inline` statement generates inline machine code which will cause the program to wait for the video controller to go into the retrace mode. Once it has determined that the video is in retrace, characters may be written to the video RAM segment without disturbing the display.

Tandy 1000 users and those with some other compatibles may ignore this code segment by commenting it out. Try it both ways — you'll be surprised at the difference it makes both in speed and visual clarity. Actual assembly code statements are provided as comments for inclusion in other programs — the assumed radix is Hex.

The procedure `define_window` draws a window area extending from the upper left corner defined to the lower right corner. The area is surrounded by a lined box and is filled with blanks in the color specified by the variable `atr`. The procedure also sets the global variable `attrib` to this value. The conditional choice here is for line type — the single-line box looks the best on the Tandy 1000 while the double-line variety looks better on the Tandy 2000.

This is not a true window in the strictest sense. No data "under" the window location is preserved and there is no provision for restoring the display data that has been overwritten. Also, you are not constrained from writing outside of the window. This is just a simple and quick method for highlighting a particular area of the video display.

The last procedure, `header`, is a simple method of writing a title line in a window. It will automatically center the text on the line below the row specified and will not overwrite the edges of the window. Text may be displayed using a different attribute from the one used to build the window (contained in `attrib`). This procedure may also be used to center text between any two column positions on a given line. Just remember that the text will be displayed *one line below* the row parameter.

The main program is a simple vehicle to demonstrate these features. After properly initializing the program values, the main program will draw 11 concentric windows on the video screen with three lines of text in them. The window attribute is changed each time a key is pressed.

Conclusion

I hope you have enjoyed this introduction into "video magic." I cannot advocate this type of programming for general programs. These techniques are best when used in specific applications. Remember, the more you use features like this, the less likely your program will ever run on another type of machine. You may have just discovered the next compatibility test!

Used well and judiciously, the effects can be astounding. *Menu* is a prime example — I feel it would be totally worthless if it took several seconds to display the screen information.

Next month, I'll continue with my exploitation of the video potential of the MS-DOS computers. I have had an outstanding response to my *Menu* articles and I thank you for your interest. □

Table 2

Tandy 2000 Monochrome Video Attributes

Attribute Byte

REVID	INT	BLINK	MS1	MS0	BLANK	BKC	BLC

Attrib Bit	Explanation
REVID	Setting this bit will display the character in reverse video.
INT	Setting this bit will display the character in the high-intensity or highlight color mode.
BLINK	Setting this bit causes the character to blink.
MS1-MS0	Normally used with MS0 = 0 to denote character mode. In this mode, MS1 controls underlining.
BLANK	Setting this bit will blank the corresponding character position.
BLC-BKC	BLC = 0 causes the cursor to blink in the corresponding position and BKC = 1 displays a reverse video block.

More information on the Tandy 2000 Video System may be obtained in the *Tandy Model 2000 Technical Reference Manual* (Radio Shack Cat. No. 26-5404, $19.95)

The listing:

```
program demo;

{*****************************************************************************

   The demonstration program contains all of the code to normally compile
   and run on the IBM-PC or any close compatible.  It will demonstrate the
   principles of direct video access by using a simple window manager to
   draw boxes containing text on the screen.

   The program contains all of the code necessary to run on either the IBM
   clones or on the Tandy Model 2000.  Where machine-dependent code is
   required, the alternate code and any comments are contained within a
   comment block set off by using the delimiters "(*" and "*)".

   To compile the program for the Tandy 2000, remove the special delimiters
   from the "2000" blocks and put delimiters around the IBM blocks of code

*****************************************************************************}


const
(*
{- TANDY 2000 ------------------------------------------------------------}
  cnormal   = $0A;              { Tandy 2000 attribute for normal video      }
```

```
  crevrs   = $8A;          { Tandy 2000 attribute for reverse video     }
  chilite  = $4A;          { Tandy 2000 attribute for hilite video      }
  crhilite = $CA;          { Tandy 2000 attribute for reverse hilite    }
  color1   = 0;            { Normal background color is Black           }
  color2   = 11;           { Normal foreground color is Bright Cyan     }
  color3   = 4;            { Hilite background color is Red             }
  color4   = 14;           { Hilite foreground color is Bright Yellow   }
  smode    = 2;            { Default screen mode is 80 X 25 monochrome  }
{-------------------------------------------------------------------------}
*)
{- IBM-PC ---------------------------------------------------------------}
  cnormal  = $0A;          { Normal video Bright Green on Black Bkgnd   }
  crevrs   = $60;          { Reverse video Black on Yellow Bkgnd        }
  chilite  = $4E;          { Hilite video Bright Yellow on Red Bkgnd    }
  crhilite = $31;          { Reverse Hilite Blue on a Cyan Bkgnd        }
  smode    = 3;            { Default screen mode is 80 X 25 color       }
{-------------------------------------------------------------------------}


type
  result =
    record
      AX, BX, CX, DX, BP, SI, DI, DS,  ES,  Flags : integer;
    end;
  str80 = string[80];

var
  upper_row,
```

Text processing power that no other program can equal.

PCSG says "Send it back in 30 days for a full refund if you don't agree."

on Snap-in™ Cartridge $99.

WriteROM©

WRITE ROM is the definitive word processing extension for the Model 100. PCSG the first text formatter for the Model 100, now sold by Radio Shack as Scripsit 100. Now 18 months later PSCG introduces WRITE ROM. Those who have experienced it say WRITE ROM literally doubles the power of the Model 100.

First of all, WRITE ROM as its name implies is on a snap-in ROM. You simply take a quarter and open the little compartment on the back of the Model 100 and press it in. It is as easy as an Atari game cartridge. You can use other ROM programs like Lucid whenever you wish.

WRITE ROM lets you do every formatting function you would expect like setting margins, centering, right justifying and having headers and footers. But it does them under function key control.

WRITE ROM remembers your favorite format settings so that you can print a document without any setup, but you can change any formatting or printing parameters instantly with a function key.

WRITE ROM's "pixel mapping" feature shows you an instant picture on the screen of how your printout will look on paper.

In all there are 64 separate features and functions that you can do with WRITE ROM, and some of these features are truly breakthroughs for the Model 100.

First, WRITE ROM lets you do search and replace. Any word or phrase in a document can be searched for and replaced with any other phrase where the search words appear.

Second, WRITE ROM lets you send any text (formated or not) to any other computer over the phone with just a function key. What's more it dials and handles sign on and sign off protocol automatically.

Third, WRITE ROM has a wonderful feature called Library that lets you record favorite phrases, words or commonly used expressions (often called boilerplate).

Any place you wish any Library text to appear you just type a code. WRITE ROM automatically inserts the text just like a Xerox Memory Writer.

Picture what you can do with that kind of capability.

WRITE ROM is blindingly fast. No one can claim faster operation. Because it is on ROM it uses virtually none of your precious RAM. It works with any printer, serial or parallel. You can make a duplicate copy of a document file under a new filename. Rename or delete (kill) any RAM file with function key ease.

This description only scratches the surface of the amazingly powerful piece of software. Dot commands allow control of such things as margins, centering, line spacing and other changes in the middle of a document. Most are Wordstar compatible.

A mail merge feature allows you to send the same document to every name on your mailing list, personalized for each recipient.

WRITE ROM enables you to do underlining, boldface and correspondence mode as well as any other font feature like superscripts that your printer supports in a way that many users say "is worth the price of the program."

To underline you don't have to remember a complicated printer code. You just type CODE U, and to stop underline, CODE U again. The CODE key is to the right of your spacebar. Boldface? CODE B to start and stop. Easy to remember and do. Five different printer features of your choice.

We couldn't list all the features here. For example, not just double space but triple or any other. You can use your TAB key in a document. WRITE ROM allows you to indent. This means you can have paragraphs that have a first line projecting to the left of the rest of the paragraph. Plus many more features.

WRITE ROM has a feature unique to any word processor on any computer. It is called FORM. FORM is an interactive mechanism that lets you create screen prompts so that you or someone else can answer them to fill out forms or questionaires.

With FORM anyplace where you had previously typed a GRAPH T and a prompt in a document, WRITE ROM will stop and you are shown that prompt on the screen. You can type in directly on the screen and when you press F8 you see the next prompt. Goes to a printer or a RAM file.

Think of how you can use FORM. A doctor or nurse could use it for a patient's history with each question appearing on the screen. An insurance salesman could have his entire questionaire. You could construct a series of prompts to answer correspondence typing the answers, even using Library codes. This feature lets you answer letters in rapid fire fashion each with personalized or standard responses.

Before WRITE ROM you had to be a programmer to create a series of prompts. Now its as simple as GRAPH T.

PCSG makes the claim that WRITE ROM is the easiest, fastest and most feature rich formatter for the Model 100. We are happy to offer WRITE ROM because it expands the 100 to a dimension of text processing you cannot equal on even larger computers.

We brashly state that WRITE ROM is the best you can buy. But put that to the test. If you aren't as excited as we are return it for a full refund. Priced at $99 on snap-in ROM. MasterCard, VISA, American Express and COD.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd., #207, Dallas, Texas 75229 • 1(214) 351-0564

PCSG provides hotline software support for the Model 100 Call us at 1-214-351-0564

```
  upper_col,
  lower_row,
  lower_col,
  scr_type,
  counter     : integer;
  ch          : char;
{****************** Constants Used By the Video Routines ******************}

  video_mem   : integer;      { Video Memory Paragraph Segment value        }
  attrib      : integer;      { Video attribute set by define_window        }
  regs        : result;       { Record definition for system communication  }
  normal      : integer;      { Video attribute                             }
  revrs       : integer;      {              values set                     }
  hilite      : integer;      {                    by init_video            }
  rhilite     : integer;      {                          determination      }

{*****************************************************************************}

procedure off_cursor;         { Turn off the cursor on the video screen     }
begin
  regs.AX := $0100;
(*
{- TANDY 2000 ------------------------------------------------------------}
  regs.CX := $6000;
{------------------------------------------------------------------------}
*)
{- IBM-PC ----------------------------------------------------------------}
  regs.CX := $3000;
{------------------------------------------------------------------------}

  intr(16,regs);
end;

{*****************************************************************************}

procedure on_cursor;          { Turn on the cursor on the video screen with }
begin                         { a flashing block                            }
  regs.AX := $0100;
(*
{- TANDY 2000 ------------------------------------------------------------}
  regs.CX := $000F;
{------------------------------------------------------------------------}
*)
{- IBM-PC ----------------------------------------------------------------}
  regs.CX := $0007;
{------------------------------------------------------------------------}

  intr(16,regs);
end;

{*****************************************************************************}

procedure cursor(x, y : integer);
begin                         { This procedure replaces the GotoXY routine  }
  regs.AX := $0200;           { in Turbo as it is based with (1,1) as       }
  regs.BX := 0;               { the upper left corner -- this routine assumes}
  regs.DX := (y shl 8) or x;{ as does the BIOS code that the upper left   }
  intr(16,regs);              { corner of the video screen is (0,0)         }
end;
```

```
{*************************************************************************}

procedure getcursor(var x, y : integer);
begin                          { Replaces the WhereX and WhereY routines in   }
  with regs do                 { Turbo and returns colum values in "x" and row}
    begin                      { values in "y" based on the upper left corner }
      AX := $Ø3ØØ;             { of the display being (Ø,Ø) unline the Turbo  }
      BX := Ø;                 { routines presumption of (1,1)                }
      intr(16,regs);
      x := lo(DX);
      y := hi(DX);
    end
end;

{*************************************************************************}

procedure mode( scrmode : integer );
begin                          { sets video screen mode accordingly           }
  regs.AX := scrmode and $ØØØF;
  intr(16,regs);
end;

{*************************************************************************}

procedure init_video;
var
  scrmode : integer;
begin                          { This procedure initializes constants and     }
  scrmode := smode;            { video ports properly for the specific hard-  }
  lowvideo;                    { ware.  For the Tandy Model 2ØØØ, the mono-   }
  clrscr;                      { chrome mode is used and color monitors will  }
  normal := cnormal;           { see colors generated on the monitors.        }
  revrs  := crevrs;
  hilite := chilite;
  rhilite:= crhilite;
(*
(- TANDY 2ØØØ ------------------------------------------------------------)
{
  The Tandy 2ØØØ color monitor will generate up to four colors in the
  monochrome mode of operation depending on the attributes written in the
  video RAM and the colors programmed to the ports listed below.

  The colors are used in pairs with the attributes:  the normal colors
  are used to form normal and reverse video displays and the hilite colors
  are used to form the hilighted text in both the "normal" and reverse
  video modes.

  The monochrom video RAM segment address is transient and depends on the
  amount of RAM installed in a particular machine.  This routine uses an
  MSDOS interrupt call to determine the total memory installed and then
  calculates the video segment address from the returned value.  This
  address is in the form yFCØH (y = 3, 7, B, etc.)
}
  mode(scrmode);
  port[$198] := color1;        { Set normal background color value            }
  port[$19A] := color2;        { Set normal foreground color value            }
  port[$19C] := color3;        { Set hilite background color value            }
  port[$19E] := color4;        { Set hilite foreground color value            }
```

Factory Authorized

Radio Shack® DEALER

Discover a Whole New World of TV Entertainment!

SATELLITE SYSTEM

U.L. Approved

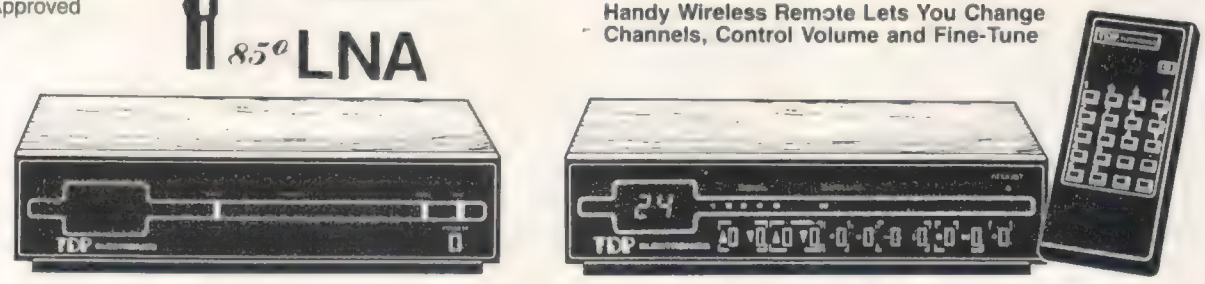
85° LNA

User Installable! Over 100 Channels! Uncut Movies! Specials! Sports! News! Weather! Music Videos!

Entire System **$1519**

- Totally Automatic System With 8½-Ft. Dish, Receiver and Wireless Remote Control
- Comes With All Accessories Needed for Complete Do-it-Yourself Installation
- Descrambler Ready

Handy Wireless Remote Lets You Change Channels, Control Volume and Fine-Tune



Now you can enjoy it all! 24-hour non-stop entertainment that you simply can't get from local TV, cable, and even video tape! You'll see the hottest stars, recent uncut and uninterrupted movies, the ultimate in sports coverage—and the best part is, there are no monthly fees to pay. You'll see exciting family-oriented programs, plus music videos, FM broadcasts and movies with provision for stereo, and 24-hour news coverage on both TV and radio. It all adds up to out-of-this-world TV viewing at a down-to-earth price. Come in today and discover how easy it is to enjoy satellite programming! Includes complete installation manual.

(817) 825-4027

NOCONA ELECTRONICS • Box 593 • Nocona, TX 76255

UPS Shipping

A RADIO SHACK DEALER

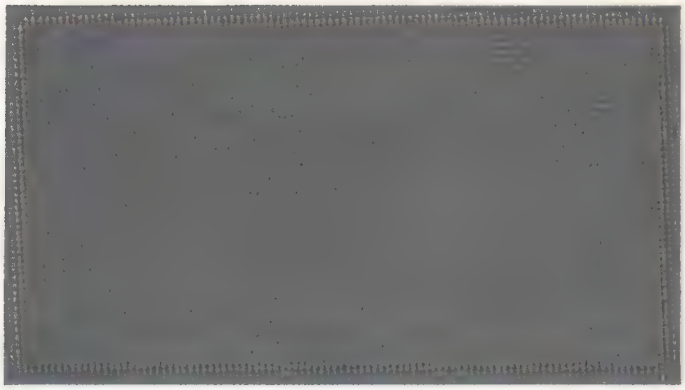
```
  intr(18,regs);
  video_mem := ( (regs.AX shl 6) and $FF00 ) or $00C0;
{-------------------------------------------------------------------------}
*)
{- IBM-PC ---------------------------------------------------------------}
  intr(17,regs);
  if (regs.AX and $0030) = $0030
    then                    { Only monochrome exists -- use mode 2        }
      begin
        video_mem := $B000;{ IBM-PC initialization is much easier and     }
        scrmode := 2;       { depends on the mode begin used - mono or    }
        normal  := $07;     { color.  This is determined by using a PCDOS  }
        revrs   := $70;     { interrupt to return the equipment status word}
        hilite  := $0F;     { for examination.                            }
        rhilite := $7F;
      end
    else video_mem := $B800;
  mode(scrmode);
{-------------------------------------------------------------------------}

  off_cursor;
end;

{*************************************************************************}

procedure disp_str(x, y : integer; strline : str80);
{
  Display the string from parameter "strline" at the screen location
  determined from "x" and "y".  The text color is determined by the
  current value of the global variable "attrib"
}
var
  offset, i : integer;
begin
  for i:= 1 to ord(strline[0]) do
    begin
      offset := (( y * 80 ) + x) shl 1;
      x := x + 1;

{- IBM-PC ---------------------------------------------------------------}
{
   Generates inline code to wait for the video monitor to be in the
   retrace mode prior to writing characters to the monitor's video
   segment addresses.  This code may not be required for all compatibles
   (in particular, Tandy 1000) and can be removed to speed up the
   character display rate.
}
        inline(
        $B8/$40/$00/            {           MOV   AX,0040   }
        $1E/                    {           PUSH  DS        }
        $8E/$D8/                {           MOV   DS,AX     }
        $8B/$16/$63/$00/        {           MOV   DX,[0063] }
        $83/$C2/$06/            {           ADD   DX,6      }
        $1F/                    {           POP   DS        }
        $EC/                    { LOOP1:    IN    AL,DX     }
        $A8/$01/                {           TEST  AL,1      }
        $75/$FB/                {           JNZ   LOOP1     }
        $FA/                    {           CLI             }
        $EC/                    { LOOP2:    IN    AL,DX     }
```

```
        $A8/$01/                    {           TEST  AL,1      }
        $74/$FB/                    {           JZ    LOOP2     }
        $B8/$0C/$00/                {           MOV   AX,000C   }
        $48/                        { LOOP3:    DEC   AX        }
        $75/$FD                     {           JNZ   LOOP3     }
        );
{-------------------------------------------------------------------------}
      memw[video_mem:offset] := ( attrib shl 8 ) or (ord(strline[i]));

{- IBM-PC ---------------------------------------------------------------}
      inline($FB);                  {           STI             }
{-------------------------------------------------------------------------}

    end;
  cursor(x,y);
end;

{*************************************************************************}

procedure define_window(urow,ucol,lrow,lcol,atr : integer);
{
  Defines a "window" marked by the upper left corner (urow,ucol) and
  the lower right corner (lrow,lcol).  The window is "drawn" on the video
  screen in the colors determined from the parameter "atr" and global
  variable "attrib" is set to these colors for future use.

  Note that this is not a true "window" in the strictest sense of the word.
```

```
  The routines in this program will right anywhere on the video and not
  just in the area defined by the window.
}
const
(*
{- TANDY 2000 ------------------------------------------------------------}
  ulc = $C9;                { Double-line graphics characters to generate  }
  urc = $BB;                { the boxes around the windows on the video    }
  llc = $C8;                { screen                                       }
  lrc = $BC;
  hrz = $CD;
  vrt = $BA;
{------------------------------------------------------------------------}
*)
{- IBM-PC ----------------------------------------------------------------}
  ulc = $DA;                { These single-line graphics characters look   }
  urc = $BF;                { better on the IBM-PC graphics display than   }
  llc = $C0;                { the double-line characters above -- your     }
  lrc = $D9;                { choice!!                                     }
  hrz = $C4;
  vrt = $B3;
{------------------------------------------------------------------------}

var
  window_line : str80;
  line_len, i, j : integer;
begin
  attrib := atr;
  line_len := lcol-ucol+1;
  window_line[0] := chr(line_len);
  window_line[1] := chr(ulc);
  window_line[line_len] := chr(urc);
  for i := 2 to line_len-1 do
    window_line[i] := chr(hrz);
  disp_str(ucol,urow,window_line);

  window_line[1] := chr(vrt);
  window_line[line_len] := chr(vrt);
  for i := 2 to line_len-1 do
    window_line[i] := ' ';
  for i := 1 to lrow-urow-1 do
    disp_str(ucol,urow+i,window_line);

  window_line[1] := chr(llc);
  window_line[line_len] := chr(lrc);
  for i := 2 to line_len-1 do
    window_line[i] := chr(hrz);
  disp_str(ucol,lrow,window_line);
end;

{************************************************************************}

procedure header(row,col,ecol : integer; str : str80; atr : integer);
{
  This procedure will write a header line below the top line of the
  current window determined from its parameters.  "row" is the top line
  of the window and "col" is the left column; "ecol" is the right column
  value.  The header text is centered in line one below "row" and will
  not overwrite the edges of the window.  The text is written in the
```

```
    colors specified by the parameter "atr"
  }
  var
    workline   : str80;
    i, k, len  : integer;
    atsave     : integer;
  begin
    atsave := attrib;
    attrib := atr;
    len := ecol - col - 1;
    for i := 1 to len do
      workline[i] := ' ';
    k := ( (len - ord(str[0])) div 2) + 1;
    for i := 1 to ord(str[0]) do
      begin
        workline[k] := str[i];
        k := k + 1;
      end;
    workline[0] := chr(len);
    disp_str(col+1,row+1,workline);
    attrib := atsave;
  end;

  {*****************************************************************************}

  begin {"demo" main program}

    init_video;

    upper_row := 0;
    upper_col := 0;
    lower_row := 24;
    lower_col := 79;

    for counter := 1 to 11 do
      begin
        case ( counter mod 4 ) of
          0: scr_type := normal;
          1: scr_type := revrs;
          2: scr_type := hilite;
          3: scr_type := rhilite
        end;

        define_window(upper_row,upper_col,lower_row,lower_col,scr_type);
        header(upper_row,upper_col,lower_col,'This is a test header',scr_type);
        disp_str(28,12,'This string is centered');
        header(lower_row-2,upper_col,lower_col,
               'Press any key to continue',scr_type);

        read(KBD,ch);
        upper_row := upper_row + 1;
        upper_col := upper_col + 1;
        lower_row := lower_row - 1;
        lower_col := lower_col - 1;
      end;

    mode(2);
    on_cursor;
  end. {demo}
```

PCM



Portable Protection

By Chris Miller

Being able to protect the files and programs in the memory of your Model 100 can sometimes be a necessity. In the office, you might want to limit the access to some or all files in the computer. You might also want to hide secret documents from prying eyes in your home.

Although the best security for your computer might be a vault, there are a few ways to protect valuable information in the Model 100's RAM using software. I have written three programs that offer different levels of security. They can be used individually or combined although you could run into memory space problems on an 8K machine if all are used at once.

The first program simply hides the name of a file from the menu. The second codes any document file or any BASIC program saved in ASCII format. By coding the file in memory, you can prevent someone from using Text to view a file or peeking into memory from BASIC.

The best combination is to use the second and third programs.

The third program provides password protection of files. A machine language program is used to keep the user from aborting with SHIFT and BREAK.

While setting up the password you can also enter five lines of text to be displayed upon power-up.

Program Operation

Listing 1, INVFIL.BA is the program that hides the filename from the menu. You may leave out the spaces while typing in the listings.

After executing INVFIL, enter the complete filespec of your choice with the extension .DO or .BA. The operation has a toggling effect on the menu. To make a file visible again, enter the filespec of an invisible file. INVFIL will also work with the five built-in programs.

If you forget the name of an invisible file, execute the following line from BASIC:

```
FOR X=63842 TO 64138:PRINT CHR$(PEEK(X));:NEXT X
```

Listing 2, FILCDR.BA codes any document (.DO) file in memory or on tape. To encode a .DO file, run FILCDR and enter the complete filespec of your choice.

You will be prompted for an eight-character code word. This code word can be created by using any combination of keys on the keyboard except for the function keys, CTRL-C (BREAK) and CTRL-M (ENTER). After typing the eighth character, you choose where the output will be stored. This operation also has a toggling effect on the output.

To decode a file for use, just enter the appropriate filespec and code word again. If you use the cassette recorder for storage, press "Record" and/or "Play" before you run FILCDR.

When coding a file in RAM, you must have enough free memory equal to the size of the file to be coded plus 550.

Warning, forgetting code words to your files can be hazardous to your health! It is very hard to decypher the type of code used.

Listing 3, PASWRD.BA is a BASIC program that creates a machine language program with user-created text and password built in.

To enter the five lines of text, execute PASWRD and type

(Chris Miller has been using Tandy computers for about three years. He is currently attending college, studying data processing and business.)

in up to 40 characters per line. Lines shorter than 40 characters will be automatically centered on their line in the final program. Any of the printable characters can be used to show information to anyone who turns on your computer such as the owner's name, address and phone number.

Once the information is typed in, the machine language program is created in memory and saved as PASWRD.CO, the IPL function is set with IPL"PASWRD.CO" and memory size is set with CLEAR 0,62653 — all automatically.

Type in the following one line program and save it as PROTCT.BA.

```
10 CLEAR0,62653:IPL"PASWRD.CO":POWEROFF
```

To activate the password program, run PROTCT.BA from the menu. If you press the PRINT key while running PASWRD.CO, you will have to press the reset button to enter the password. Be sure to have a copy of PASWRD.BA if you want to change the text and password in the future.

The same warning as given for FILCOR also applies to PASWRD. Forgetting the password can mean loss of access to the programs and files in your computer. Should this happen and you didn't use FILCOR, you can do a cold start and use the PEEK function to scan memory reloading programs and files.

Two hints that can help save you from any accidents:

- Practice using the programs only after making backups of any important files and programs.
- Keep a list of passwords and code words that go with each coded file somewhere away from where your computer is used. □

Listing 1:

```
Ø REM INVFIL.BA written by Chris Miller
1Ø CLS:CLEAR1ØØ:PRINTCHR$(27);"V";:PRINT
@13,"Invisable File":PRINT:FILES
2Ø PRINT@28Ø,;:LINEINPUT"Enter complete
filespec ; ";FI$:IFMID$(FI$,LEN(FI$)-2,1
)<>".","THENFI$=FI$+STRING$(7-LEN(FI$),32)
+CHR$(Ø)ELSEFI$=LEFT$(FI$,INSTR(FI$,".")
-1)+STRING$(9-LEN(FI$)," ")+RIGHT$(FI$,2
)
3Ø FORIN=63845TO64131STEP11:FORX=ØTO7:CK
$=CK$+CHR$(PEEK(IN+X)):NEXTX:IFFI$<>CK$T
HENCK$="":NEXTINELSEPOKEIN-3,PEEK(IN-3)X
OR8
4Ø MENU
```

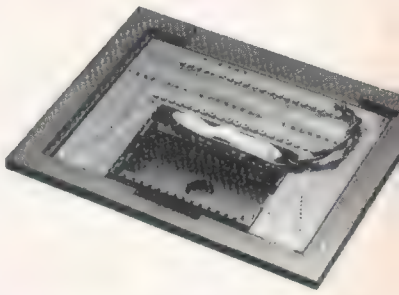
Listing 2:

```
Ø REM FILCDR.BA written by Chris Miller
1Ø CLS:MAXFILES=2:PRINT@1Ø,"File Protect
ion"
2Ø PRINT@4Ø,;:INPUT"Enter full Filespec:
 ";FI$
```

Give your Model 100

128K RAM

Installs as easily as plugging in a socket



Software included, transfers from bank to bank. Works like main menu! Includes powerful RAM Basic that lets programs store and access data from any other bank.

PCSG says: Satisfaction guaranteed or your money back within 30 days

As amazing as it seems you can upgrade your Model 100 to 128K of RAM in just 60 seconds.

It comes to you right out of the box looking just like the picture. You just open the little compartment on the back of your Model 100 with a quarter and it just pushes right into place. You can then put the cover back in its place.

You then have 4 banks of RAM of 32K each. The additional three banks also work just like your Main Menu.

You push a function key and you are in the second bank. Push again and you are in third, again, then fourth. Press it once again for your original bank.

It has its own built-in NiCad battery that recharges right from the Model 100 and its guaranteed for a full year.

What is really great is that you can copy a file from one bank to another with just a function key.

Each bank is like having another Model 100, and all the built-in programs as well as any snap-in ROM programs appear in all four banks and work the same way. Your widebar cursor moves from file to file and you access any file or run any program just by pressing ENTER.

What lets you copy any file from one bank to another is a snap-in ROM from PCSG called RAM+, that comes at no extra charge. It just pushes right into the little socket in that same compartment with the 96K expansion unit.

Not only does this firmware let you copy a file from bank to bank, but you can make a copy of any file within the same bank instantly with a function key. Great for Lucid spreadsheets!

Copy a file from bank to bank with a function key

You can also rename a file, or kill any file with just a function key. Plus you can do a whole lot of other useful things like setting the date, day and time with function key ease. You even have a function key that lets you use non-Radio Shack printers without having to make those tricky dipswitch settings.

RAM+ lets you cold start any one of your banks without affecting the other three. That means that anytime you want you can clean out a bank's entire memory, but leave intact all the files in the other banks.

What is also fantastic is that you don't have to have the ROM in place to use the additional RAM. Whenever you take out the snap-in ROM it leaves behind a tiny machine code program that lets you switch from bank to bank just by pressing ENTER.

This lets you use your ROM socket to snap-in other ROMS like LUCID spreadsheet, WRITE ROM text processor, or DISK+ ROM file transfer program, and use them in any or all four banks. All of these, by the way, are available from PCSG.

When you are ready to copy a file from one bank to another or use any of the other fantastic functions we talked about you can just snap the RAM+ ROM back into place.

Everybody that has this 128K system in their Model 100 is so excited, because it gives them four times the capacity and all banks work just like the Main Menu.

And what has made a lot of people happy is that the system bus, located in the same compartment, is left free for you to plug in a DVI or the Holmes Engineering/PCSG portable disk drive.

The ability to copy a file from bank to bank instantly with a function key, plus all of the other features make this RAM extension truly an engineering masterpiece.

Some people hesitate when they think of installing something, and then others are skeptical that any additional hardware could be as good as the Model 100 itself. That's why we sell these 96K expansions on a 30 day trial. Simply return it within 30 days for a full refund if you are not satisfied. Priced at $425. MC VISA COD.

1-214-351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd. No. 207, Dallas, Tx. 75229

VISA / MASTER CARD / AM. EXP. / C.O.D.

```
3Ø PRINT@8Ø,"Enter 8 character code: ";:
FORI=1TO8:I$=INPUT$(1):CO$=CO$+I$:PRINTI
$;:NEXTI
4Ø PRINT@12Ø,"1>MEM to CAS, 2>CAS to MEM
, 3>MEM to MEM";:TR=VAL(INPUT$(1))
5Ø IFTR=1THENOPENFI$FORINPUTAS1:OPEN"CAS
:"FOROUTPUTAS2
6Ø IFTR=2THENOPEN"CAS:"FORINPUTAS1:OPENF
I$FOROUTPUTAS2
7Ø IFTR=3THENOPENFI$FORINPUTAS1:OPEN"ZZZ
ZZZ"FOROUTPUTAS2
8Ø IFNOTEOF(1)THENPRINT#2,CHR$(ASC(INPUT
$(1,1))XORASC(MID$(CO$,(DMOD8)+1,1)));:D
=D+1:GOTO8Ø
9Ø CLOSE1,2
1ØØ IFTR=3THENGOSUB13Ø:GOSUB12Ø
11Ø MAXFILES=Ø:MENU
12Ø KILLFI$:NAME"ZZZZZZ.DO"ASFI$:GOTO11Ø
13Ø OPEN"ZZZZZZ.DO"FORINPUTAS1:OPENFI$FO
ROUTPUTAS2
14Ø IFNOTEOF(1)THENPRINT#2,INPUT$(1,1);:
GOTO15ØELSECLOSE1,2:RETURN
```

Listing 3:

```
Ø REM PASWRD.BA written by Chris Miller
1Ø CLS:PRINTCHR$(27)"V";:CLEAR65Ø,62653:
PRINT@47,"Password Protection Setup"
2Ø PRINT@12Ø,"Enter up to 4Ø characters
for title line"
3Ø FORX=1TO5:PRINT@2ØØ,"Line ";X:LINEINP
UTA$(X):A$(X)=STRING$(INT(4Ø-LEN(A$(X)))
/2,32)+A$(X):A$(X)=A$(X)+STRING$(4Ø-LEN(
A$(X)),32):PRINT@24Ø,STRING$(4Ø,32);:NEX
T
4Ø B$=+A$(1)+A$(2)+A$(3)+A$(4)+A$(5)+"Pa
ssword > "+STRING$(6,Ø):FORX=62743TO6295
9:POKEX,ASC(MID$(B$,X-62742,1)):NEXT
5Ø PRINT@12Ø,STRING$(55,32);:PRINT@2ØØ,;
:PRINT"Enter the new Password > ";:FORX=
1TO8:R$=INPUT$(1):PRINTR$;:P$=P$+R$:NEXT
:FORX=62654TO62661:POKEX,ASC(MID$(P$,X-6
2653,1))XOR39:NEXT
6Ø H$="Ø123456789ABCDEF":FORX=62662TO627
42:READA$:POKEX,(INSTR(1,H$,LEFT$(A$,1))
-1)*16+(INSTR(1,H$,RIGHT$(A$,1))-1):NEXT
7Ø DATA 21,53,F6,36,8Ø,21,AA,FF,36,ØØ
8Ø DATA CD,31,42,Ø6,C8,21,16,F5,CD,ED,1B
,CD,22,42,Ø6,ØC,21,DE,F5,CD,ED,1B,Ø6,Ø9,
11,A8,F9,CD,CB,12,12,1C,Ø5
9Ø DATA CD,2Ø,ØØ,78,FE,Ø1,C2,EB,F4,ØE,Ø9
,21,A8,F9,11,BE,F4,1A,EE,27,47,7E,B8,C2,
C6,F4,2C,1C,ØD,79,FE,Ø1,C2,Ø2,F5,C3,97,5
7
1ØØ IPL"PASWRD.CO":SAVEM"PASWRD.CO",6265
4,62959,62662
```

PCM

Hardisk Accounting Series Has Six Modules, Many Extras

Software 1000/1200/2000/3000

The Great Plains *Hardisk Accounting Series* consists of six stand alone modules: *General Ledger*, *Accounts Receivable*, *Accounts Payable*, *Inventory Control*, *Job Cost* and *Payroll*. Information from each separate program integrates into the *General Ledger*. These accounting programs work on a Tandy 2000 with a 5 MB hard disk with 128K RAM or on the model 1200.

These programs are powerful but complicated. They are meant for a large company that is considering a mainframe, but wants to keep the same information on a microcomputer. Before the system is installed, the user needs to "size" the program. Sizing is a combination of unique parts, parts received and sold per week, serial numbers, number of parts for which a sales history will be maintained, plus 400K for the system files.

Inventory Management with point-of-sale invoicing comes with four floppy disks which must be installed on a hard disk. Some of the features of version 3.02 that improve inventory management are tracking of inventory items by serial number, and support of LIFO, FIFO and Weighted Average valuation.

The program produces inventory turnover reports and assists with "floor planning" by use of a serial number aging report. Price lists with up to five price levels and three quantity breaks are produced, and 15-character alphanumeric part numbers are accommodated.

The program offers a choice of printing sales slips or invoices. Substitutions for out-of-stock items are suggested. Up to three different taxes per customer are calculated automatically and customized sales invoice formats can be created.

Cost sales, account sales, returns and quotes are handled, and a screen inquiry for items in six user-defined categories can be customized. Items are suggested in LIFO or FIFO order to assist in proper rotation of inventory, and the software links with a cash drawer to perform as a cash register.

Productivity is increased because clerical work is reduced. Discounts and quantity prices breaks by customer are calculated automatically. When integrated with the Great Plains *General Ledger* module, duplicate entries are eliminated. Promotional sales reports are produced to assist managers in evaluating the effectiveness of advertising strategies. A great feature is the physical inventory checklist, which reduces auditing time.

The program comes with a step-by-step tutorial, which is necessary for such a powerful accounting package. To fully learn the system, about two hours is needed. However, once the tutorial is completed, the user is ready to begin using the program.

The software is password protected to restrict unauthorized hackers access to the company's inventory. The user's manual has a flowchart for the inventory program and another for the integrated system.

The point-of-sale invoice function is very useful. When a sales slip has been generated, the quantity on hand is automatically reduced from inventory. Sales slips and quotes are numbered sequentially from 1 to 999999, although the number can be manually overridden at the time of sale.

The sales portion recognizes four types of transactions: cash, sale on account, a quote or a return. If an item is out of stock, the program adds a back order and updates the purchase receipts file. If *Inventory Control* is integrated with *General Ledger*, a three-digit code for each "profit center" is added to the accounting information.

When the master file is set up, standard default prices are indicated. If the clerk wants to discount, either by a percentage or a dollar amount, this is easily done. State tax is added by default, or can be manually input. Four payment options — cash, check, credit card or on account — are accepted. If the module is integrated with *Accounts Receivable*, the new customer is assigned a seven-digit number and is added to the mailing list after the invoice is prepared.

Many useful reports are produced by the inventory program: inventory item list, price list, inventory stock status report, sales analysis by product category and purchasing advice reports. Other management information includes on order and back order reports, inventory receivings lists and voucher registers.

Detail sales history, quotes history, detail sales reports and quotes reports are also produced. Other reports produced show detail by salesperson, inventory sales, sales journal, GL distribution account register, duplicate document register or inventory checklist. Item inquiry by user category, promotion sales report, purchase receipts report, serial number aging report, inventory turnover report, serial number history report and inventory posting account list are also produced.

The *Job Cost* module is a very precise program. A two-hour tutorial helps understand the system. The software records labor, equipment, material, overhead, transit, maintenance and miscellaneous expenses with three levels of detail. The whole job, or components may be estimated. Customer information for each job is stored. The program acts as a mini-spreadsheet and allows "what if" questions. Changes to estimates may be saved or ignored.

The *Job Cost* module can stand alone or be integrated with the other modules. Unbilled costs are handled. Contracts

can be changed and the computer-calculated bill amounts could be overridden if necessary. Three types of billing — fixed price, cost plus or cost plus with a maximum can be handled.

Profit may be calculated as a percentage or as a flat rate for each aspect of the job. Accounting is made easier by integration with *Accounts Receivable*, *Accounts Payable*, *Payroll* and *Inventory Control*. A complete audit trail is maintained to show all adjustments to estimated and actual costs. The retainage amount on each bill is automatically calculated.

Each type of expense is assigned a code. The system tracks expenses for estimates and jobs using these codes. Once a code is assigned, it remains in the system making subsequent job costings much easier. The only time new codes are needed is when a new expense occurs.

The actual entry of job costs consists of defining "phases" or the highest level of expense breakdown. Optional "steps" and "details" provide more information about the project. The only limit to the number of phases, steps or details is the capacity of the hard disk. The cost accounting part of the program requires detailed estimates of the phases, unless subcontracting is part of the project.

Fluctuations in the cost of labor, materials or climate variations may make revision of an estimate necessary. Changes are made at the detail level and the step and phase levels are automatically adjusted. The estimate projection feature allows the estimator to experiment with variables. If one laborer charges $10 an hour and another one $20, but works three times as fast, the program prints two reports based on the salary variable.

After the estimate is accepted by a customer, the file is transferred to a job file. Once work begins on a job, a record of expenses is kept. If changes in the contract are necessary after the work has begun, the existing contract can be modified. The source of increases in expenses is clarified, and an audit trail identifies the changes.

A pre-billing report indicates which expenses and profits combine to produce a billable amount. The contractor indicates the percentage of work completed, and the system suggests an amount to bill. This amount can be defaulted, or another amount based on the completion/cost ratio can be input.

Reports produced by the *Job Cost* program are a labor table list, estimate edit report, list of current estimates, job bid report, job cost labor transaction journal and a job cost accounts payable transaction register. Invoice expense by account is detailed in the distribution register. The jobs in process report includes all current jobs, amounts billed and received to date, projected cost, actual cost to date and estimated cost of completion.

Other reports detail work outstanding, job audits, profit and loss, variance analysis, job reference and job status. Accounting information is reported in the memorandum entry, pre-billing summary, job cost billing, general ledger distribution and job cost cash receipts journal. At the end of an accounting period, a year-end closing and overhead applied report show the costs, billings and net profit for the year, as well as the amount of overhead applied to each job and the percentage of total overhead from open jobs.

It takes a bit of practice to learn the *Inventory Control* and *Job Cost* programs, but the information and management reports produced by both

modules is extremely useful. One of the nicest features is the constant inventory update, which triggers reorder reports when the inventory levels are low.

These accounting modules are well worth the price because so many extras are included. Construction and manufacturing companies will be pleased with the ease of use and the practical accounting features of this software.

(Great Plains Software, 1701 Southwest 38th Street, Fargo, ND 58103; 701-281-0550; $695 per module.)

— M.J. Batham

Book

Dot Matrix Manual A 'Rare Find'

The computer section of any large bookshop is crammed with books. To the novice, most are tomes too complicated to be practical and helpful. To those more familar with microcomputers, many of the books lack sufficient information to justify the expense. It is rare to come across a computer book that is both reasonably priced and full of useful information. The *Minute Manual for the DOT Matrix Printer* is one of those rare finds. Its author, Jim Pirisino, has put together a relatively small volume (164 pages) that is written for those who are thinking about purchasing a dot matrix printer. The book is logically organized, providing a wealth of information in the least amount of space.

The first chapter briefly describes the various types of printers: letter quality, thermal and dot matrix. It also lists the advantages and disadvantages of each and concludes that for most general applications, including speed and price, the dot matrix is hard to beat. Also, the introduction describes, with good illustrations, what a dot matrix printer is and how it works. In order to better explain the operations and features of a dot matrix printer, the author chose six different printers to feature: Epson FX, Gemini 10X, Okidata 92, Prowriter, NEC 8023A and the Apple Imagewriter. As explained, these were selected because of their popularity and the author's experience. Also, the price range of these full-featured machines was attractive: $250 to $599.

In the second chapter, printing features of the dot matrix printer are defined and discussed. Such features as normal, enhanced, double strike and bold print modes are covered as well as special features such as character width, alternate character sets (foreign symbols), backspacing, underlining, subscript, superscript, etc. Each feature is given a single page of thorough description and a simulated print example. At the bottom of the page is a legend where all six of the selected printers are listed and the ones supporting the feature are indicated by filled blocks — a nice touch.

Chapter 3 discusses the most generally touted aspect of dot matrix printers by salespeople — speed. The most popular measure of speed is characters per second. This standard gauge is not, however, the whole story. The author explains that determining speed involves many factors: whether the printer is bidirectional (prints in both

directions), how unfilled lines are printed, whether the printer has a storage buffer, and the amount of time it takes for the machine to advance the paper one line. In addition, using special features such as bold or double strike modes requires the printer to make a second pass over previously printed characters, requiring more time. As an example, the six printers were tested for speed. A chart is given showing the results, which the author terms "the practical printing speed." A total of 26,332 characters were printed. While the printers boasted various speeds from 120 to 160 characters per second, the practical speeds ranged from a low of 54 to a high of 100 characters per second. The author's results are enlightening, and show how printing speed can be overrated without considering other aspects of the printer.

In ending the chapter, five topics are briefly covered that are necessary in choosing a printer: full service (by the dealer and manufacturer), warranty, durability, ribbons, noise and forms handling. Here, many of us might temporarily forget these concerns while being prodded by an eager sales representative or being too anxious to get the printer home. Seasoned buyers may find this digression by the author a bit naive, but experience has proved that such information is critical when a printer sits broken and useless.

The fourth chapter discusses how printers are connected to a computer. It covers serial and parallel cables and the advantages and disadvantages of both, including photos showing six different types of cables. Text processing, graphics and fonts are briefly presented, describing popular hardware interfaces which support these printing features. Also, there is an informative section on printing high resolution graphics.

Chapter 5 describes the hows of using a printer. The author begins by presenting a rating system based on the degree of interaction between programs and printers. The rating system is based on five levels of interaction:

Rating 0: "The program does not allow a printer to be used at all."

Rating 1: "The program allows the use of a printer, but no printer codes can be sent to the printer while the program is being used."

Rating 2: "The program will allow you to send printer codes to the printer while operating the program, but no printing codes can be sent while printing is in progress."

Rating 3: "The program will allow you to send printer codes to the printer *before or during* the printing so that different styles of print can be generated in the same document."

Rating 4: "The program will allow you to change print styles before or during printing and allows you to customize or is already customized for your particular printer."

Each rating is thoroughly discussed. This is an excellent rating system for evaluating both programs and printers.

The remainder of the chapter covers, in detail, the ASCII code and how computers and printers use the code in generating characters and controlling various printer functions. Though this section may be beyond the novice, it is, nevertheless, an excellent presentation of how the computer and the printer communicate. Moreover, the two types of computers, 7-bit and 8-bit, are discussed and why the differences between the two affect the way their programs interact with a printer. Two complete charts are also given which illustrate the various graphics characters generated for each of the six featured printers operated by the 7-bit and 8-bit computer. This is valuable information for those interested in graphics.

The fifth chapter closes with a discussion of how BASIC is used to print characters and send control codes to a printer. Examples are given for the Apple, IBM PC, and Commodore 64 computers. The hexadecimal system is also introduced because, as the author explains, some printers require that hexadecimal be used to invoke printer codes. Fortunately, most of the popular printers do not have this requirement.

Chapter 6 is a continuation of chapter 5. It presents information about forms handling, such as form feed, top of form, line spacing, line feed, forward, reverse, carriage return, skip over perforation, reset and horizontal and vertical controls. Each function is defined and explained with illustrations showing how each function is used in practical applications. Also, the different types of paper feed mechanisms are discussed: pin, friction and tractor feeds.

In chapter 7, each of the six featured printers — Epson Fx, Gemini 10X, Prowriter, NEC 8023, Okidata 92 and the Apple Imagewriter — are compared, feature for feature and function for function. Detailed information is given along with examples of the different print fonts available for each machine. By the time you have finished this chapter, you should be thoroughly familiar with the dot matrix systems presented. It is a fascinating comparison study.

The last chapter is a buyer's guide. Blank forms are given in which all the functions previously covered are listed with adjacent spaces where you can check off the features you need. These forms will be helpful in the process of seeking and evaluating a dot matrix printer that will fit your exact requirements, with no guess work.

Not only is the *Minute Manual for the DOT Matrix Printer* an excellent guide, but it is also a handy reference for programmers and users who already have a dot matrix printer and want to take full advantage of its capabilities. I recommend this manual as part of your computer library.

(MinuteWare, P.O. Box 2392, Columbia, MD 21045, $12.95)

— Ralph Rideout

Software **1000**

Starclash II: Strategy for Star Warriors

The sector is still this evening; no hum of enemy engine nor whine of stardrive disturbs the night. The view from your craft is that of a universe at peace. Yet, as supreme commander of the Argolian Polity, you are uneasy. Where are the 50 enemy ships from Dormst headed? Have you taken all reasonable precautions? Are your home defenses adequate? Will your ships sent to the Fex planetary system be victorious? Should you attack the productive Gallum, or the enemy's home Harnor, or the relatively undefended Opel? And what of Cabble? Of Jeral? What about the World-egg??

You turn and shift through your intelligence reports — a 12 percent chance of victory? You sigh deeply. Perhaps you are too old for this command, too old to dispatch the young ones to battle. You will not be person-

ally involved in the action, of course. No motor skills are needed here. Your task is that of strategy, of plotting, of studying the gathered information, of anticipating and dashing the plans of the enemy. A sudden noise outside draws your attention. Looking out quickly, you now know where the 50 ships from Dormst were headed.

This is the star-warring scenario of *Starclash II* by Baen Software, a $30, one-or-two-player strategic simulation for the 256K Tandy 1000. It is not copy-protected. The software, in an attractive package, has a 24-page booklet containing eight pages of a sci-fi scenario, "The Story So Far," a wryly humorous introduction, clear instructions and a section explaining how to customize saved sequential game files. My teenage son and several younger neighbors had no problem with the directions. If you skim the scenario and plunge into the instructions, within five minutes total confusion will reign; within 10 the light begins to dawn; and in less than 15 you will have lost your first universe, muttered something about "Aha! — that's their game!", and be planning your strategy for another session.

At the start of each game, you are first given the option of loading a previously saved one. If you elect a fresh start, you are asked to select the number of star systems involved (two to 20), two- or three-dimensional complexity, the number of human players (one or two), names for each player and the number of extra star systems for each player (0 to 18, depending upon the number of star systems chosen). This last parameter serves as a handicapping system. The program then configures the systems (each game is different), builds a transit table between systems and randomly determines initial strengths and other factors. You are given a chance now, as you are at the end of each cycle, to save the current game to disk with a one-line identifier. The players then enter their movements alternately, and action continues until one side is eliminated.

The general goal is to quickly inhabit as many star systems as possible, particularly the more productive ones, which create many additional ships each cycle. Once units are launched to another system, they cannot be recalled. Combat occurs when both sides are present at a star system and is resolved by the program with random results. Experienced simulation players might desire more control at this stage than the program provides.

The main screen is the General Report screen, which displays the Situation Report, the Economics Report, a quick view of the sector and the available commands (F1 - F10). The command keys make it possible to receive complete details on any one system, to display how many units are in hyperspace, to see travel times between systems, to move your units and to see an estimate of your chances of winning. The most interesting of these is F2, the Display Tank, which enables the arrow keys to rotate the display through different viewing planes in higher level games in order to better understand the third dimension used.

I've noted that different players soon adopt favorite command keys when planning strategy, since much of the needed information is available from more than one source. Planetary security reasons preclude discussion of my particular usage of these.

Depending upon the conditions specified at the beginning, a game may take from 10 minutes to over two hours to complete. The great majority of this time will be your own planning, since

the computer's moves and displays are completed almost instantaneously.

On the plus side, the program functions quickly and flawlessly, although it is a bit cocky — once my fleet outnumbered the computer's by a margin greater than 4 to 1, yet my odds of winning were only 22 percent!

I do have a few minor complaints with *Starclash II*. First, there is no use of color or sound. I realize that this is a strategy simulation, not an arcade game, but judicious use of color would make sections of the displays clearer. Also, a few times I was merrily entering my deployments when a glance at the screen showed "No movement" because of a typing error. A short warning beep here would be advantageous. I also think a more elaborate finish is called for. After spending one hour and 47 minutes saving the universe, a flashing "Pat Wins" is hardly a sufficient reward.

Second, there is no way of completing a game without totally eliminating the enemy. If you have 3000 ships and the enemy has three, the game continues. Strategic surrender of either side should be an option, especially when the odds of winning drop below 10 percent (which they do quite often), though my vengeful, rather blood-thirsty son disagrees with me. As a side thought, perhaps a time-limit version could be added, with percentage of accomplishment.

Last, the two-player option is fun, but it seems interminable. The player not moving must turn his head or take a walk for several minutes each turn. Few of the local space warriors recruited for review service cared for this mode after a game or two.

Overall, the local consensus was "It's a nice little game, but I don't think I'd spend 30 bucks on it." If you enjoy strategic simulations, though, you could do worse.

(Baen Software, 8 West 36th Street, New York, N.Y., 10018, 212-947-8244, $29.95)

— Pat Pugliano

Software **1000/1200/3000**

Chart Master: Quality Graphs for MS-DOS Users

One of the greatest pleasures a computer user can have is to find a useful program that works exactly as stated on the first attempt. *Chart Master* from Decision Resources is such a program. There are, however, a couple of stumbling blocks for Tandy users that I will get into later.

Chart Master is a program for producing presentation quality graphs for MS-DOS machines. It features support for on-screen, printer, and plotter displays, as well as slide creation.

The Version 6.1 release I received for review is bound in a sturdy three-ring binder. All documentation is printed on a high quality paper with very readable print. Instructions are clear and accurate, with many illustrations to show the user exactly what should be happening. It is obvious that a great amount of thought and effort has been put into producing this manual, and many other software suppliers could benefit from studying the style of this document. As is popular among MS-DOS applica-

tions, the manual is stored in a booklike casing for convenience.

The program comes on a set of copy-protected diskettes. Decision Resources sends you a second copy of the program upon registration. Installation on a fixed disk is supported in two variations. The user may install the program in a manner making it "fixed disk" bootable only once. Additionally, installation files for fixed disk setups in which the original must be in Drive A, are provided. Hardware requirements include an MS-DOS compatible computer with minimum of 192K of memory (256K required for charts to be produced on a printer).

Program installation for your particular combination of equipment is easily performed following manual and on-screen instructions. It was at this point that I found the first problem for Tandy users having a floppy drive setup. Instructions are given for copying DOS onto the program disk. Tandy's MS-DOS version is simply too large to fit in the disk space remaining. The only solutions are to use PC-DOS or boot up from a DOS disk prior to running *Chart Master*. Questions on the monitor are clearly presented for many optional setups. Screen colors for palette background, drive configuration, port assignments, plotter and printer installation questions follow.

It is with printer installation that the Tandy user may run into another snag. Although large numbers of printers are supported, there is not a single Tandy/Radio Shack printer listed. A call to Decision Resources was made and I was informed that no current plans include adding Tandy printer selections. Of course, some of the selections available are compatible with various Tandy printers, but many users will not be aware of which ones they are. If you have other than a Tandy printer, you should not find any difficulty in printer selection.

At last, with program installation complete, we come to the ultimate question: How does it work?

The answer here is a resounding *excellent*! I was able to run through the five-minute example in just about that amount of time, easily producing a high quality chart on my Toshiba 1340. I then spent the rest of the day making up charts of everything I could think of. Bar charts, line charts, pie charts . . . you name it! . . . I tried it. In no instance did I have any problem other than those caused by my own oversight. The charts produced are outstanding. The user can optionally select vertical or horizontal charts, choose the plot area, add footnotes, print data "on the plot" or not, produce regression curves, even mixed-style charts. Color choices are given, hatching patterns (what the bars, etc. are filled with) can be user selected, and on and on and on. During chart creation there are no less than eight fonts available with 16 sizes for each. Fonts include the standard, bold, Roman, bold Roman, script, Gothic, Swiss and symbol. The symbol font allows the inclusion of many mathematical and scientific symbols, arrows, trademark and registration marks, and even cattle, people, boats, airplanes, etc., all with a single keystroke.

A feature called "Datagrabber" allows charts to be created from other application program files. After trying out the sample file, I loaded up a *Lotus 1-2-3* file previously saved as a print file. Using the cursor to grab the data I wanted to chart, I was able to quickly convert the file into a useful graphic printout.

Chart Master offers support for a multitude of plotters. Unfortunately, I don't have a plotter and consequently was unable to use this capability. Decision Resources has provided a comprehensive documentation section covering each supported plotter. Reading through several installation procedures, I found them to be easy to follow and plotter users should not have difficulty in getting set up.

Additionally, the Polaroid Palette Recording System is supported for those users needing slide presentation of their charts.

On the occasion I had to call Decision Resources, I found the staff to be courteous and knowledgeable. The few questions I had concerning the program were answered easily and professionally. I suspect users will find the support of the product more than adequate.

To sum it up, I find the program to be excellent in all respects. If you have a need for charting application, I certainly would not hesitate to recommend this one. For Tandy printer owners, I would suggest you do a bit of research as to compatibility before you make the program purchase.

(DecisionResources, Inc., 25 Sylvan Road South, Westport, CT 06880, 203-222-1974, $375)

— Leonard Hyre

The Management Advantage: Silver Lining With Small Cloud

The Management Advantage by Thoughtware is designed to help the user become an efficient personnel manager. The hardware requirements list includes IBM functionally compatibles with MS-DOS 2.0, 256K and two double-sided disk drives. I received the program to review on a Tandy 1200HD (MS-DOS 2.11, 256K), but as the hardware requirements list also includes IBM XT, I expected no compatibility problems. I was dissappointed in this regard: more on this later. Highly recommended by the documentation are a color/graphics adapter, a color monitor and a printer. My system includes a graphics adapter with monochrome monitor, which seemed to work just fine with the program. I share the recommendation for a printer to obtain hard copies of the sessions.

The documentation is very professional: three-ring binder in a hard sleeve, attractive and well designed. This leads the user through the necessary installation procedure which, in the case of the Tandy 1200HD, creates a directory on the hard disk complete with the necessary files. Installation also involves the user supplying information about self and subordinates. Once this is complete, the documentation is basically augmentative: the program is menu-driven and easy to use.

The main menu (called the Thoughttree) consists of three choices: "Consult with Management Advantage," "Update Subordinate Information," or "Print Summary of Last Consultation."

A consultation session with *The Management Advantage* is designed to represent a conversation with your own personnel management consultant. As the orientation suggests, the benefit of the program is that it allows you to "think out loud" about managing your people. The software guides you through this process with questions. Your answer determines the next question, and thus the path through the session to the solution to the problem at hand. This question-answer-question sequence is really the heart of the program. The documentation cites impressive credentials in management and behavioral sciences as a basis for the principles underlying the methods utilized.

The Management Advantage allows three distinct approaches to consultation. You can think through a plan involving a particular subordinate in detail. If you feel you need a better grasp of the principles underlying a particular management task, this approach is also available. If you have already pinpointed the problem you are having with a particular employee, the approach to take is trouble-shooting.

If you need assistance at any point during a session, a help menu is only a keystroke away. Using this feature you can obtain background on the topic or definition of a term, then continue where you left off. The help appears in a window — the frame remains until you've rejoined the main stream of the consultation session.

"A single keystroke produces a professional, personalized report"

At appropriate points during a session, you are allowed the option of making notes. These notes are printed as a part of the summary. I like this feature. It might prevent your brilliant revelation from slipping away.

At the end of each consultation you will be offered the option of a printed summary of the session. A single keystroke produces a professional, personalized report for your files, outlining the problem or task, complete with an action plan detailing the solution at which you have arrived. Whether you choose to print the summary or not, the report is saved on the disk, and can be printed the next time you use *The Management Advantage* by making that selection from the "Thoughttree."

The third selection possible from the "Thoughttree" allows you to supply information about subordinates, whom you didn't introduce during the initialization procedure.

The Management Advantage, while easy to use, is a very complex program. In several sample consultation sessions,

I was impressed by the diversity of questions posed, as well as by the number of distinct culminations reached. The question-answer segment is thought provoking. If you take the time to think through each point as it is raised, you can only increase your management skills. The action plans produced seem logical and beneficial. In the hands of the novice or moderately skilled manager, this program could produce exceptional results. It is intricate enough to also benefit the skilled manager.

This silver lining, however, has its cloud. After installing the program as instructed, my attempt to start yielded garbage from my printer and locked up my computer. I had several conversations and trial and error sessions with a helpful technician at Thoughtware before he arrived at this theory: In the Tandy 1200HD version of MS-DOS, some files that must be transferred during initialization are too long to fit on the program disk. The following "chewing gum and bailing wire" suggested by Thoughtware was necessary to make the program fly. I copied the files `ANSI.SYS` and `COMMAND.COM` from MS-DOS, together with a file called `CONFIG.SYS` from *The Management Advantage* Disk A, onto a blank disk. On this same disk, I created an empty `AUTOEXEC.BAT` file. This disk must be in Drive A at power-up. Replacing this with program Disk A and typing `AUTO EXEC` starts (or should I say: "jump starts"?) the program. Program Disk A (after initialization) includes the two MS-DOS files mentioned, and has 64K free, which makes me wonder about their diagnosis. However, their solution works. Thoughtware finally got tired of hearing from me, threw up their hands, and said they really didn't know why there is a compatibility problem. Neither do I. While this start-up procedure is not overly difficult, and I imagine a simpler, more elegant, home grown solution could be found, simpler yet would be for Thoughtware to produce a Tandy 1200HD compatible version of their program. This would remove a troublesome flaw from an otherwise excellent program. The $249 price tag on this program seems a bit much, and I certainly couldn't recommend it to 1200HD users with this flaw.

(Thoughtware, Inc., 2699 S. Bayshore Dr., Coconut Grove, FL 33133, $249)

— Stanley Townsend

Software **1000/1200/2000/3000**

Accounting Partner: Modulized Help for Small Business

The fully integrated *Accounting Partner* from Star Software (version 1.22A-1) bundles four modules in one package: *Accounts Payable, Accounts Receivable, General Ledger* and *Payroll*. Any of the individual programs can be used alone, or all can be linked together through the *General Ledger*.

The non-copy-protected program runs in monochrome on a color monitor. The software works on the Tandy 2000, 1000 or 1200 with 128K, two floppy disk drives (or optional hard disk), and a printer with 132 column capacity. Standard printed forms for invoices, checks, credit and debit memos and purchase orders are supported by the software.

The package is designed for busi-

nesses doing up to $5 million in annual sales and having under 20 employees. If the business grows, files can be converted to the *Accounting Partner II* system without reinstallation of the data.

Up to 10,000 vendors are supported by the *Accounts Payable* module. Reports are based on the "balance forward" method of accounting. Vendor information is input into the system, and all activity can be retrieved and printed on vendor lists, worksheets, purchase orders and internal management reports. Up to five different prices per item can be keyed for each vendor, thereby supporting both retail and wholesale pricing.

Vendors can be accessed by an alphanumeric account number, or by company name. Other useful features are the accommodation of nine-digit ZIP codes, phone numbers with four-digit extensions, and shipping, as well as "sold by" addresses.

Current balances for 30, 60, or 90-day aging are kept, and as each monthly statement is completed, balances are moved automatically to the next category. The program keeps track of vendor discounts, monthly interest rates for overdue balances, account terms and pricing codes. A tax resale number is listed for the vendor, in case of sales for resale which are not taxable under state laws.

The purchase order function keeps track of purchase order number, vendor and sales order number, order terms, delivery date, FOB location, shipping instructions, item numbers and description, unit price, quantity, percent discount, sales tax, freight charges and miscellaneous charges and comments.

Purchase orders are printed for a single account using either number, name or order number. Once purchase orders have been entered into the system, they cannot be changed. This ensures proper audit controls to prevent fraud.

Fixed periodical payments such as rent or insurance premiums are accommodated. The auto-invoice program searches the vendor master file and displays each account that has an auto-invoice recorded. The invoices can be paid for each vendor, or certain vendors can be skipped over for the month.

Several management reports are produced by the software, such as vendor lists, account status, payable aging and vendor status reports. At the end of the period, the data is backed up, accounts payable are aged and balances are brought forward. When all reports are prepared, the information is transferred to the *General Ledger* module.

Checks are not printed from the *Payable* module. The program can be used as a stand-alone for management purposes, but the actual checks are paid from the *General Ledger* section. One of four cash accounts is credited for the amount specified, and as many as five debit entries are posted.

The *Accounts Receivable* program prints invoices, statements and credit memos. You may access 10,000 customers by either number or name. Wholesale and retail pricing are supported, as well as volume discounting. Customer order history is shown and a sales analysis report, by product invoiced, is produced.

A master address can be set up for the firm's home office, but a secondary "shipping" detail will keep track of where the products were sent. Date and amount of last payment is detailed, as well as year-to-date payments, current balance, and 30-60, 61-90, or over-90-day aged amounts due.

Other customer facts are maintained such as credit limit, auto-invoice amount, discount, monthly interest on overdue accounts, account terms, pricing code, salesperson and resale number.

Invoices are printed for a specific amount, or the auto-invoice function can be used for fixed periodical amounts due. Editing is easily done before invoices are printed, and invoice numbers are automatically assigned.

Three 45-character description lines for the item being ordered/invoiced provided ample space for detail. Another nice feature is the use of an "item file" when invoicing. When the system is initialized, frequently sold products can be set up in a special file. A 10-digit alphanumeric code is listed, and a three-line description with five different unit prices is shown. As the customer is invoiced, typing the item number brings up the invoice description and the correct price for that customer.

Standard discount per customer is calculated automatically, non-taxable sales for resale are noted, and freight charges, state tax and other miscellaneous charges are shown on the invoice. The invoice can be printed immediately or saved into an invoice batch file to be run at the end of the day.

Once invoices are entered into the system, they cannot be changed, thereby ensuring proper audit controls. Credit memos are printed in the same manner as invoices. All payments are applied to the oldest customer balance.

New invoices and customer statements are printed by customer, or in numeric order or a user-defined order. Finance charges for overdue accounts are calculated automatically, but the data operator can surpress the addition of finance charges.

Management reports produced by the *Receivable* module are customer lists, customer account status and an accounts receivable aging report. At the end of the accounting period, a general ledger control report is printed, accounts receivable are aged into the proper category, an item file report is produced and the balances are brought forward.

Star Software's menu driven *Accounting Partner* is very easy to operate. Function keys are not needed, so memorization is unnecessary. However, the comma represents a "yes" and a period for "no," so it's faster to use these keys, rather than jumping back to the typewriter keyboard.

Customers/vendors/accounts are accessed by name, not just by number. The manuals, although well written, are not necessary since the screen shows all possible options. Editing is done by field number so modification of a record is simple and quick.

The accounting transfer to general ledger takes time; a small business with few customers, products, employees, accounts or vendors could get by with the Tandy 1000 with floppy disks. For more speed, the Tandy 2000 with the faster 80186 chip or Tandy 1200 with hard disk would be better. The optimum computer for accounting software would be the Tandy 2000 with hard disk.

The price is reasonable for the complete system. All four modules come bundled, so as the business grows you may want to integrate the system to save accounting time, rather than using only one of the programs. The *Accounting Partner* offers many features not found on other, faster programs.

(Star Software Systems, 20600 Gramercy Place, Suite 103, Torrance, CA 90501, 213-538-2511, $395 for four modules)

— M.J. Batham

New Products

The following products recently have been received by PCM, examined by our magazine staff and approved for the PCM *Seal of Certification*, your assurance that we have seen the product and have ascertained that it is what it purports to be. This month the *Seal of Certification* has been issued to:

The Banner Machine, a program for designing and printing signs and large banners from a computer printer. Requires Tandy 1000, 1200, 2000 or 3000. *Cardinal Software, 13646 Jefferson Davis Highway, Woodbridge, VA 22191, (703) 491-6502, $14.95.*

C-Num, a utility for renumbering BASIC programs. Requires Model 100. *Queue Software Systems, 4528 Belleview, Suite 210, Kansas City, MO 64111, $19.95.*

C-Sort, a machine language utility for sorting RAM data files. Requires Model 100. *Queue Software Systems, 4528 Belleview, Suite 210, Kansas City, MO 64111, $24.95.*

Campaign Math, introduces election process and teaches ratios, fractions and percentages. Requires Tandy 1000, 1200 or 3000. *Methods & Solutions, Inc., 82 Montvale Avenue, Stoneham, MA 02180, $39.99.*

Common C Functions, a book containing a library of C functions designed to be used in the reader's own programming projects. *Que Corporation, 7999 Knue Road, Indianapolis, IN 46250, (317) 842-7162, $17.95.*

dBase III Advanced Programming, a book for programmers working in the Ashton-Tate's *dBase III* database management system. Discussion on system designing and structured programming techniques. *Que Corporation, 7999 Knue Road, Indianapolis, IN 46250, (317) 842-7162, $22.95.*

Introduction to Computer Mathematics, designed primarily for classroom use, this by Russell Merris offers a practical and entertaining approach to the field of computers and mathematics. *Computer Science Press, Inc., 1803 Research Boulevard, Rockville, MD 20850, (301) 251-9050, $27.95.*

Miracle, transforms your computer into an integrated telecommunication workstation combining the ability to share data with other computers along with state-of-the-art spreadsheet technology, graphing and word processing. *Micro-Systems Software Inc., 4301-18 Oak Circle, Boca Raton, FL 33431, (305) 391-5077, $299.*

PC Paintbrush, a free-hand graphics drawing program. Requires Tandy 1000, 1200 or 3000, color graphics and mouse, joystick or digitizer. *Z-Soft, 1950 Spectrum Circle, Suite A495, Marietta, GA 30067, $139.*

RoboMath, a robot-filled arcade game that teaches multiplication and division. Requires Tandy 1000, 1200 or 3000. *Methods & Solutions, Inc., 82 Montvale Avenue, Stoneham, MA 02180, $39.99.*

Symphony Tips, Tricks and Traps, designed as a quick reference to Lotus *Symphony*, this book demonstrates shortcuts, offers help with problems and suggests ways of using some of *Symphony's* little-known capabilities. *Que Corporation, 7999 Knue Road, Indianapolis, IN 46250, (317) 842-7162, $21.95.*

Wizardry, a fantasy role-playing game involving magic, castles and a cast of characters. Uses graphics. Requires Tandy 1000, 1200 or 3000. *Sir-Tech, 6 Main Street, Ogdensburg, NY 13669, $59.95.*

By awarding a *Seal*, the magazine certifies the program *does exist*, but this *does not* constitute any guarantee of satisfaction. As soon as possible, these hardware or software items will be forwarded to PCM's reviewers for evaluation.

Using *BAREAD 2.1*

Bar code listings must be read in numerical order beginning with Line 1 and continuing through the last line of the listing. The computer display is used to prompt you as to which line to scan and give you warning messages should you happen to get out of step.

When you run *BAREAD*, it asks you to scan the first line of the bar code listing. This line contains the name of the program as well as the beginning of the program itself. The computer will sound a high-pitched beep whenever it's ready for you to scan a line. After a line has been successfully read, you'll hear a lower beep. A "blip-bloop" sound prompts you to turn your attention to the screen for a message. You'll hear this when you accidentally scan a line out of sequence.

After reading the first line, you continue scanning with the second line. Remember to wait for a high beep before scanning and then listen for a low beep to indicate a successful read.

Once the last line of the listing has been scanned, *BAREAD* will return control to the Tandy 100/200 menu screen. Note that the program you just scanned is now in the directory with a .DO extension.

The final step is to convert the .DO text file to a normal BASIC program. This is done quite simply by going to BASIC and loading the file with a command such as LOAD"TEST.DO" (if the program name were TEST). The program will load into BASIC and will be ready to run. To save the program in BASIC's compressed format (.BA extension), you'd type SAVE"TEST" (if the program were named TEST). You may then kill the .DO file with KILL "TEST.DO".

BAREAD 2.1

```
1000 ' *** Initialize ***
1010 ON ERROR GOTO 1040
1020 CLEAR 1000:MAXFILES=2
1030 GOTO 1050
1040 IF ERR=5 THEN RESUME NEXT
1050 ON ERROR GOTO 0
1060 RUNM "B3OF9"
1070 OPEN "WAND:" FOR INPUT AS #1
1080 UC%=-1
1090 PC$="0123456789ABCDEFGHIJKLMNOPQRST
UVWXYZabcdefghijklmnopqrstuvwxyz- $+"
1100 DIM RW$(36)
1110 ER$(1)="You must scan line 1 first
"
1120 ER$(2)="You've SKIPPED a line!"
1130 ER$(3)="You've ALREADY SCANNED this
 line!"
1140 ER$(4)="Code not PCM2/39 format!"
1150 ER$(5)="Command not applicable here
!"
1160 ER$(6)="You cannot skip this line!"
1170 ER$(7)="Selected resume file not in
 computer!"
1180 ' *** Read Reserved Words List ***
1190 DATA BEEP,CLEAR,CLOSE,DATA,DEFDBL,D
EFINT,DEFNG,DEFSTR,ELSE,GOSUB,GOTO
1200 DATA INKEY$,INPUT,INSTR(,LCOPY,LEFT
$(,LINE(,LOADM,LPRINT,USING,MAXFILES
1210 DATA MID$(,NEXT,PEEK,POKE,POWER,PRE
SET(,PRINT,READ,RESTORE,RETURN,RIGHT$(
1220 DATA SOUND,SPACE$(,STRING$(,THEN
1230 FOR I%=1 TO 36:READ RW$(I%):NEXT I%
1240 ' *** Procedure Begins Here ***
1250 CLS:PRINT@44,"PCM Bar Code Program
Reader v2.1"
1260 LINE(20,4)-(219,18),1,B:LINE(22,6)-
(217,16),1,B
```

```
127Ø NN%=1
128Ø GOSUB 166Ø:IF ER%>Ø THEN GOSUB 162Ø
:GOTO 128Ø
129Ø IF LL%=Ø AND INSTR("YN",IL$)>Ø THEN
 ER%=5:GOSUB 162Ø:GOTO 128Ø
13ØØ IF LL%=Ø THEN ON INSTR("ALSR",IL$)
GOTO 182Ø,189Ø,198Ø,2Ø5Ø
131Ø IF LL%=1295 THEN 135Ø
132Ø IF LL%<>NN% AND NN%=1 THEN ER%=1:GO
SUB 162Ø:GOTO 128Ø
133Ø IF LL%<NN% THEN ER%=3:GOSUB 162Ø:GO
TO 128Ø
134Ø IF LL%>NN% AND NN%>1 THEN ER%=2:GOS
UB 162Ø:GOTO 128Ø
135Ø IL$=RIGHT$(IL$,19)
136Ø IF LL%=1 AND NN%>Ø THEN GOSUB 178Ø
137Ø CL$=CL$+IL$
138Ø FOR I%=1 TO LEN(CL$)
139Ø     CH$=MID$(CL$,I%,1)
14ØØ     IF CH$="%" THEN GOSUB 151Ø:IF NL
% THEN 147Ø ELSE GOTO 144Ø
141Ø     IF CH$="/" THEN GOSUB 155Ø:IF NL
% THEN 147Ø ELSE GOTO 144Ø
142Ø     IF CH$="." THEN UC%=NOT(UC%):GOT
O 145Ø
143Ø     IF CH$=>"A" AND CH$<="Z" AND NOT
(UC%) THEN CH$=CHR$(ASC(CH$)+32)
144Ø     XX$=XX$+CH$:IF RIGHT$(XX$,1)=CHR
$(13) THEN PRINT#2,XX$;:XX$="":UC%=-1
145Ø NEXT I%
146Ø CL$=""
147Ø PRINT@2ØØ,SPACE$(8Ø);
148Ø IF LL%<>1295 THEN NN%=LL%+1:GOTO 12
8Ø
149Ø ' *** Done ***
15ØØ CLOSE:CALL 618Ø7!:CLEAR 5ØØ,HIMEM:M
ENU
151Ø ' *** Decode Reserved Word ***
152Ø NL%=Ø:IF I%>LEN(CL$)-1 THEN NL%=-1:
CL$="%":GOTO 154Ø
153Ø I%=I%+1:CH$=RW$(INSTR(PC$,MID$(CL$,
I%,1)))
154Ø RETURN
155Ø ' *** Decode Hex and Control Charac
ters ***
156Ø NL%=Ø:IF I%>LEN(CL$)-1 THEN NL%=-1:
CL$="/":GOTO 161Ø
157Ø I%=I%+1:IF INSTR("/%.",MID$(CL$,I%,
1))>Ø THEN CH$=MID$(CL$,I%,1):GOTO 161Ø
158Ø IF I%>LEN(CL$)-1 THEN NL%=-1:CL$=RI
GHT$(CL$,2):GOTO 161Ø
159Ø HX$=MID$(CL$,I%,2):CH$=CHR$((INSTR(
"Ø123456789ABCDEF",LEFT$(HX$,1))-1)*16+I
NSTR("Ø123456789ABCDEF",RIGHT$(HX$,1))-1
)
16ØØ I%=I%+1
161Ø RETURN
162Ø ' *** Error Codes ***
163Ø SOUND 5ØØØ,1Ø:SOUND 8ØØØ,1Ø:SOUND 5
ØØØ,1Ø
164Ø PRINT@22Ø-.5*LEN(ER$(ER%)),ER$(ER%)
;
165Ø RETURN
166Ø ' *** Get Code Line ***
167Ø PRINT@173,"";:PRINT USING "Scan lin
e ###";NN%
168Ø IF NN%=-1 THEN PRINT@173,"Scan any
line":GOTO 17ØØ
169Ø SOUND 5ØØ,5
17ØØ INPUT#1,IL$:ER%=Ø
171Ø FOR I%=1 TO LEN(IL$)
172Ø IF MID$(IL$,I%,1)="!" THEN MID$(IL$
,I%,1)=","
173Ø NEXT I%
174Ø IF LEN(IL$)<>1 AND LEN(IL$)<>21 THE
N ER%=4:RETURN
175Ø IF LEN(IL$)=1 THEN LL%=Ø:RETURN
176Ø LL$=LEFT$(IL$,2):LL%=(INSTR("Ø12345
6789ABCDEFGHIJKLMNOPQRSTUVWXYZ",LEFT$(LL
$,1))-1)*36+INSTR("Ø123456789ABCDEFGHIJK
LMNOPQRSTUVWXYZ",RIGHT$(LL$,1))-1
177Ø RETURN
178Ø ' *** Open Program File ***
179Ø PN$=LEFT$(IL$,6):IL$=RIGHT$(IL$,LEN
(IL$)-6)
18ØØ OPEN PN$ FOR OUTPUT AS #2
181Ø RETURN
182Ø ' *** Abort ***
183Ø BEEP:BEEP:BEEP
184Ø PRINT@2Ø9,"ABORT!  Are you sure?";
185Ø INPUT#1,AN$
186Ø IF INSTR("YN",AN$)=Ø THEN BEEP:PRIN
T@251,"Scan 'YES' or 'NO'":GOTO 185Ø
187Ø PRINT@2ØØ,SPACE$(8Ø);
188Ø IF AN$="Y" THEN CLOSE:KILL PN$+".DO
":GOTO 149Ø ELSE GOTO 128Ø
189Ø ' *** Skip Line ***
19ØØ IF NN%=1 THEN ER%=6:GOSUB 162Ø:GOTO
 128Ø
191Ø BEEP:BEEP:BEEP
192Ø PRINT@21Ø,"SKIP!  Are you sure?"
193Ø INPUT#1,AN$
194Ø IF INSTR("YN",AN$)=Ø THEN BEEP:PRIN
T@251,"Scan 'YES' or 'NO'":GOTO 193Ø
195Ø PRINT@2ØØ,SPACE$(8Ø);
196Ø IF AN$="Y" THEN NN%=NN%+1
197Ø GOTO 128Ø
198Ø ' *** Stop & Save ***
199Ø BEEP:BEEP:BEEP
2ØØØ PRINT@2Ø7,"STOP & SAVE! Are you sur
e?";
2Ø1Ø INPUT#1,AN$
2Ø2Ø IF INSTR("YN",AN$)=Ø THEN BEEP:PRIN
T@251,"Scan 'YES' or 'NO'":GOTO 2Ø1Ø
2Ø3Ø PRINT@2ØØ,SPACE$(8Ø);
2Ø4Ø IF AN$="Y" THEN 149Ø ELSE GOTO 128Ø
```

```
2050 ' *** Resume ***
2060 IF NN%<>1 THEN ER%=5:GOSUB 1620:GOT
O 1280
2070 PRINT@254,"Resume Mode";
2080 NN%=1:GOSUB 1660
2090 IF LL%=0 THEN ER%=5 ELSE IF LL%<>1
THEN ER%=1
2100 IF ER%>0 THEN GOSUB 1620:GOTO 2060
2110 PN$=MID$(IL$,3,6)
2120 ON ERROR GOTO 2140
2130 OPEN PN$ FOR INPUT AS #2:GOTO 2170
2140 RESUME 2150
2150 CLOSE #2
2160 ER%=7:GOSUB 1620:GOTO 1270
2170 CLOSE #2:OPEN PN$ FOR APPEND AS #2
2180 NN%=-1:GOTO 1280
```

FILCDR (FROM PAGE 78)

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
```

PASWRD (FROM PAGE 78)

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
```

28 
29 
30 
31 
32 
33 
34 
35 
36 
37 
38 
39 
40 
41 
42 
43 
44 
45 
46 
47 
48
49 
50 
51 
52 
53 
54 
55 
56 
57 
58 
59 
60 
61 
62 
63 
64 
65 
66 
67 
68
69
70

71
72
73
74
75
76
77
78
79
80

INVFIL (FROM PAGE 78)

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

Abort

Skip Line

Stop & Save

Resume

Yes

No

The Rackseller

ALABAMA	
Birmingham	Jefferson News Co.
Florence	Anderson News Co.
Madison	Madison Books
ARKANSAS	
Fayetville	Vaughn Electronics/Radio Shack
ARIZONA	
Phoenix	Computer Pro
	Tri-Tek
Tucson	Anderson News Co.
CALIFORNIA	
Citrus Heights	Software Plus
Half Moon Bay	Strawflower Electronics
Livermore	Software Galeria
North Hollywood	Levity Distributors
Santa Rosa	Sawyer's News
Sunnyvale	Computer Literacy
DELAWARE	
Middletown	Delmar Co.
Wilmington	Normar Inc.—The Smoke Shop
FLORIDA	
Boca Raton	Software, Software, Inc.
Ft. Lauderdale	Software Connection
N. Miami Beach	Almar Bookstore
Orlando	Book Mania
Panama City	Computer Systems Group
Pensacola	Anderson News Co.
Tallahassee	Anderson News Co.
GEORGIA	
Atlanta	Guild News Agency
ILLINOIS	
Chicago	Chas. Levy Distr.
	Kroch's & Brentano's Jackson Street
	Kroch's & Brentano's Wabash Ave.
	Prairie News Agency
Oakbrook	Kroch's & Brentano's
INDIANA	
Highland	Computer Health Consultants
Scottsburg	Radio Shack of Scottsburg
IOWA	
Ames	Myers TV & Stereo Repair
KANSAS	
Wichita	Lloyd's Radio
KENTUCKY	
Louisville	Hawley-Cooke Bookstores
Prospect	Falsoft, Inc.
LOUISIANA	
Gretna	Computer Supply Store
Slidell	Radio Shack
MASSACHUSETTS	
Littleton	Computer Plus
Marlboro	Radio Shack
MICHIGAN	
Petosky	B & L Sound, Inc.
Rochester	Rochester Book Center
Sterling Heights	Software City
	Programs, Inc.
Wyoming	Gerry's Book Co.
MISSOURI	
Independence	Digital Enterprises, Inc.
St. Louis	Software Center
NEBRASKA	
Lincoln	Hobby Town
NEW HAMPSHIRE	
West Lebanon	Verham News Corp.
NEW JERSEY	
Clinton	Micro World II
Hackensack	Total Circulation Service
Pennsville	Dave's Electronic Radio Shack
NEW MEXICO	
Alamogardo	New Horizons Computer Systems
Albuquerque	News and Bookstore
	Page One Newsstand
NEW YORK	
Hudson Falls	G.A. West & Co.
NORTH CAROLINA	
Cary	News Center in Cary Village
Sylva	Total Data Systems
NORTH DAKOTA	
Fargo	Computer Associates
OHIO	
Cincinnati	Cinsoft
Toledo	Leo's
OKLAHOMA	
Oklahoma City	Merit Computers
PENNSYLVANIA	
Philadelphia	Newsy
Phoenixville	Stevens' Radio Shack
RHODE ISLAND	
Newport	Kelly's Variety
TENNESSEE	
Knoxville	Anderson News Co.
Memphis	Computer Center
Nashville	Mosko's Book Store
Smyrna	Delker Electronics, Inc.
TEXAS	
Bay City	Impulse Computers
Ft. Worth	A & A International
VIRGINIA	
Alexandria	Alonso Book & Period
Norfolk	I-O Computer
Wytheville	Wytheville Electronics
WASHINGTON	
Kennewick	Satellite TV Specialists
Mt. Lake Terrace	Emerald Computer
WISCONSIN	
Ladysmith	Electronics, Etc.
Madison	20th Century Books
ARGENTINA	
Cordoba	Informatica Y Telecomunicaciones
CANADA:	
ALBERTA	
Calgary	Rainbow Software Services
Edmonton	CMD Micro
	Kelly Software Distributors
BRITISH COLUMBIA	
Sidney	Sidney Electronics
QUEBEC	
LaSalle	Messageries De Presse Benjamin
NOVA SCOTIA	
Halifax	Atlantic News

Also available at selected B.Dalton Bookseller stores

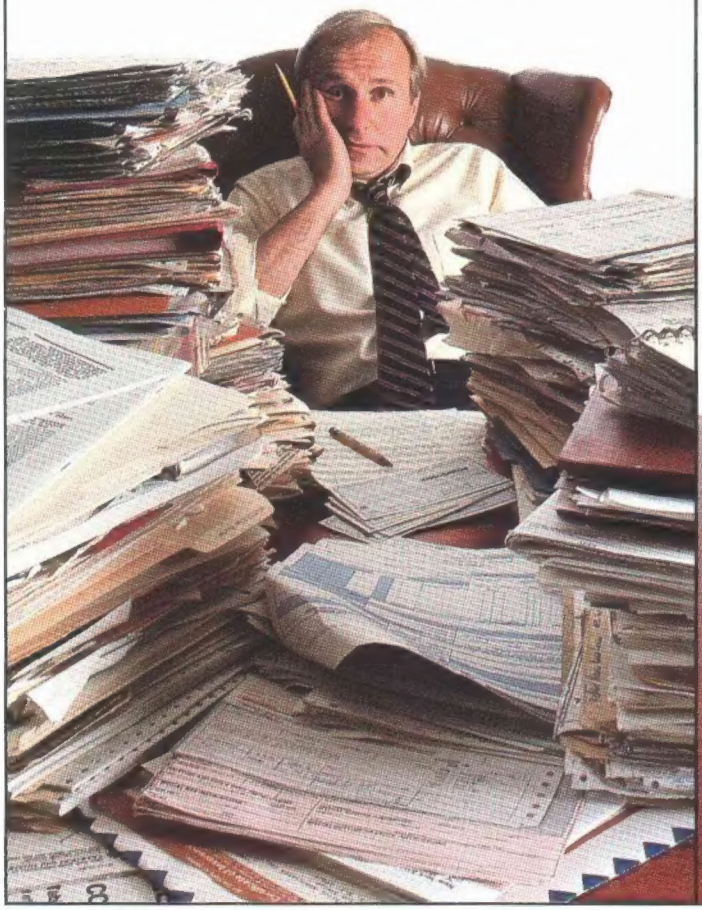
Advertisers Index

PCM Advertising and Marketing Representatives

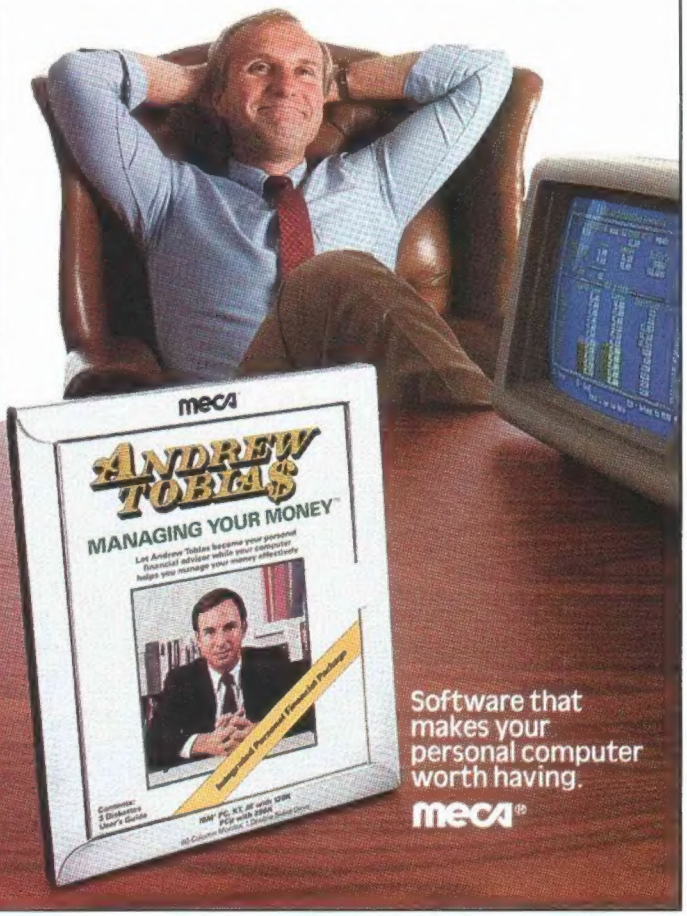
Cindy Shackleford
Director, West Coast Office
12110 Meridian South — Suite 8
P.O. Box 73-578
Puyallup, WA 98373-0578
(206) 848-7766

Kim Vincent
Advertising Representative
The Falsoft Building
P.O. Box 385
Prospect, KY 40059
(502) 228-4492

MANAGING YOUR CHECKBOOK?
MANAGING YOUR BUDGET?
MANAGING YOUR BILLS?
MANAGING YOUR CASH FLOW?
MANAGING YOUR TAXES?
MANAGING YOUR INSURANCE?
MANAGING YOUR STOCKS?
MANAGING YOUR BONDS?
MANAGING YOUR REAL ESTATE?
MANAGING YOUR TAX SHELTERS?
MANAGING YOUR SAVINGS?
MANAGING YOUR MORTGAGE?
MANAGING YOUR AUTO LOAN?
MANAGING YOUR RETIREMENT?
MANAGING YOUR CALENDAR?
MANAGING YOUR CHARGE ACCOUNTS?
MANAGING YOUR CAPITAL GAINS?
MANAGING YOUR ANNUITIES?
MANAGING YOUR APPOINTMENTS?
MANAGING YOUR DIVIDENDS?
MANAGING YOUR INTEREST?
MANAGING YOUR RECORDS?
MANAGING YOUR VALUABLES?
MANAGING YOUR KEOGH'S?
MANAGING YOUR IRA'S?



MANAGING YOUR MONEY®
BY ANDREW TOBIAS.
THE ONLY FINANCIAL
SOFTWARE THAT DOES IT ALL.



Software that
makes your
personal computer
worth having.

meca®

SUPER ROM

Lucid Spreadsheet Write ROM Database Outliner

NOW AVAILABLE ON THE M-200

NOW YOU CAN REALLY HAVE IT ALL!

All on one ROM. Truly the finest four programs available for the Model 100 — guaranteed. Try it for 30 days. If you aren't blown away by the excellence return it for a full refund.

$199.95

The four best programs for the Model 100 all on one ROM. 32K of power without using any RAM for program storage. This is the PCSG Snap-In ROM that just presses easily into the little ROM socket in the compartment on the back. You access the four right from the main menu like built-ins.

Write ROM — the definitive word processor for the Model 100. Function key formatting or dot commands. Search and replace. Library feature — inserts words, phrases or whole documents into text from just a code. MAP lets you see a picture of your document. In all there are 60 features and functions. No one can claim faster operation. FORM lets you create interactive forms with on-screen prompts that you can answer from the keyboard. Nothing else for the Model 100 compares with the features of Write ROM. Exactly the same as the Write ROM sold as a single program. Infoworld says it "makes the Model 100 a viable writing unit . . . surpassed our highest expectations for quality and clarity."

Lucid Spreadsheet: This is the one PICO magazine says "blows Multiplan right out of the socket" and Infoworld performance rated as "excellent" and said "makes the Model 100 compute." Gives you features you cannot get with Lotus 123. Lets you build spreadsheets in your Model 100 that would consume 140-150K on a desktop. Program generating capability with no programming knowledge required. Variable column widths. Includes find and sort with function key control. It's fast, recalculates like lightning. No feature has been taken from the original, only new ones added.

Database: This is a relational data base like no other. You can do everything from mailing lists to invoices. No complicated pseudo-coding, you create input screens as simply as typing into TEXT. You are not limited by size; you can have as large an input screen as you wish. Prints out reports or forms, getting information from as many files as you like. Complete math between fields. Total interface with Lucid worksheets.

Outliner: Does everything that Thinktank does on a PC but a whole lot better. Includes a Sort for your headlines. Lets you have headlines of up to 240 characters. Has cloning, hoisting and sideways scroll up to 250 characters. Like Lucid, this one sets a new standard for outliners. This is the way to plan and organize your projects.

Present Lucid and Write ROM owners can upgrade for $100. If you have both it's $75.

As usual PCSG sells the Super ROM on a thirty day guarantee. If for any reason you are not satisfied, simply return it for a full refund.

We are excited about this product. Super ROM gives the Model 100 the true power of a desktop. No other multi-program ROM has software that compares. But don't take our word for it. We invite you to make that comparison yourself. Priced at $199.95 on Snap-In ROM.

Got stuck with somebody else's multi-ROM? We'll upgrade it for $100.

(214) 351-0564

PORTABLE COMPUTER SUPPORT GROUP

11035 Harry Hines Blvd., #206, Dallas, TX 75229

MC, Visa, American Express, Check, or C.O.D.